COMPUTERS IN CHEMICAL EDUCATION: A STUDY INVOLVING SYMMETRY AND CRYSTALLOGRAPHY

A thesis
submitted in partial fulfilment
of the requirements for the Degree
of
Doctor of Philosophy in Chemistry
in the
University of Canterbury
by

R. S. Temple

University of Canterbury
1985

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

In order to determine ways in which computer assisted learning techniques may be applied with advantage to particular aspects of chemical education, software development was undertaken to produce programs capable of using a number of these techniques. The software, all items of which employed considerable use of graphics, included lecture demonstration programs, programs for auto-elaborative use, an "individual use" tutorial program and drill and practice revision programs. Most significant aspects of the teaching of selected individual topics from initial introduction in lectures to revision before final examination could be covered using programs selected from those developed.

The general subject area chosen was symmetry and crystallography since this was considered particularly suitable for graphics oriented teaching methods, an area of strength for computers. Complete teaching packages, incorporating dual purpose demonstration and auto-elaborative programs with drill and practice revision lessons, were written for the topics of basic symmetry operations and space groups. Demonstration programs with varying degrees of auto-elaborative character were also developed for the Patterson function, point groups, unit cells and Bragg's law.

An essential aspect of the project was an attempt at an evaluation of the teaching effectiveness of the material

produced. This was done through the use of concept difficulty and attitude surveys administered to the students in the classes the programs were used in. The results would suggest that development of integrated packages of various types of teaching software is both possible and worthwhile in chemical education.

CHAPTER I

INTRODUCTION

The purpose of the research undertaken in this project was to investigate the applicability of computer assisted learning (CAL) techniques to chemistry by the development and trial of computer software to aid in the teaching of a specific topic area. The emphasis was to be on the use of graphics for while some research has been undertaken on the use of computer graphics to teach aspects of subjects in science [1], little work has been done in chemistry to systematically study and utilize the potential of computer graphics in a teaching role. With the advent of affordable microcomputer systems with graphics capabilities there is now no barrier to the development of effective educational software which may used by large numbers of chemists. A specific objective was the development of a program "package" to deal with several topics within a broad subject area and to cover these topics from first introduction through to final revision.

1.1 NATURE OF RESEARCH PERFORMED

The exploration of methods by which teaching software in chemistry may be made most effective has been carried out by means of the development of a series of computer programs designed to teach selected aspects of symmetry in chemistry.

These programs, which make up the bulk of this project, follow the teaching of two selected topics, elementary symmetry operations and space groups, from first exposure in lectures through to final revision before examination. The teaching strategy was initially to use the computer as a demonstration aid in lectures and then subsequently to allow individual students, outside lecture times, to use the same programs to further explore the subject partly guided by supplied exercises. For both topics sets of drill and practice revision programs were prepared and made available to the students.

In the case of the elementary symmetry operations topic a tutorial program, designed to be used on an individual basis, was written as an alternative to lecture presentation of material and made possible a comparison of the two approaches [2].

Further programs, covering other topics within symmetry and crystallography, were also written concentrating on lecture demonstrations and subsequent auto-elaborative use by students. It was considered more advantageous to concentrate on the development of such programs rather than produce a complete "start to finish" package for each specific topic, since although the programs written required only matching drill and practice programs to complete the package, such question and answer programs are often time consuming to develop and offer little new insight since they have had considerable study in the past. On the other hand comparatively little work has been done on demonstration

programs and most of that has concentrated on either graphical plots or diagrams of actual apparatus.

The joint topics of symmetry and crystallography were selected as the area for the development since the spatial aspect of much of the subject creates difficulties when using traditional teaching methods. As with chemistry in general, this field has been the subject of a number of isolated teaching programs, some of which make very good use of graphics, on both small and larger computer systems. However little effort appears to have been made to translate these into an integrated set.

When undertaking any CAL development it is important that its effectiveness is tested in some manner. A concept difficulty survey was undertaken initially to determine areas of greatest difficulty within the subject area and later to find any changes as a result of using computers during the teaching process. Attitude surveys were used before and after the teaching course to determine student opinion with respect to computers in education.

1.2 CHAPTER SUMMARY

Background for this work is given in chapters two and three. Chapter Two is devoted to the history and application of CAL and explains the terms used to describe CAL programs as well as giving a comparison between computer based teaching and the other techniques currently in common use. Chapter Three deals with the aspects of chemistry

being taught by describing which areas of symmetry and crystallography are covered by this study and some of the methods currently employed to teach them.

Chapter four deals with the results of a crystal structure determination undertaken for the purposes of providing experience in the subject and potential teaching material. A number of the ideas for the programs developed had their origins in this study and these are explained in this chapter together with details of the structure determined.

The choice of the hardware and software systems used during the project are the subjects of chapters five and six respectively. These choices are important as they are the subject of much debate in computer and education circles and the final choice affects the ease with which programs are written, the performance of the programs and who can ultimately use the programs.

Chapters seven through eleven cover the characteristics and usage of the teaching programs. Each chapter corresponds to a single major program or group of programs.

Utility programs used, programming techniques used, and adaptations to available hardware are covered in chapter twelve.

The results of surveys of student opinion and comprehension undertaken during the course of the project are detailed in chapter Thirteen.

The packaging and presentation of programs is dealt with in appendix A.

CHAPTER II

COMPUTER ASSISTED LEARNING

Computer Assisted Learning (CAL) may be broadly defined as the use of a computer as a learning resource where some or all of the teaching material in a given course is stored and presented by the computer [1]. Within this definition there is scope for many types of teaching strategies, the choice of which depends on the nature of the teaching material, the recipients of the material and the available resources. Examination of the available types of CAL and their possible utility in a particular subject area is the first step when planning to use computers in the teaching of that subject. When investigation of CAL was undertaken for this project emphasis was placed on those aspects most amenable to the use of graphics since the subject area is largely spatial in nature and therefore suited to teaching strategies based on diagrammatic displays.

2.1 TYPES OF CAL

CAL programs may be classified according to the manner in which the teaching material is presented and the type of effect they have. Many variations of such classifications are possible [3], but the one which proved most convenient is made up of the following six categories:

1:tutorial,

2:drill and practice,

3:simulation,

4:demonstration,

5:games,

6:auto-elaborative.

1: Tutorial programs involve the presentation of information to the student where control over that presentation is directed from within the program. In the simplest case this involves a "programmed learning" approach [4] involving the presentation of a series of "pages" of information where the order of presentation is pre-determined in the program with no input from the student. In more genuine tutorial programs the control of presentation and the course of the lesson is determined by student input. This input is usually in the form of answers to questions, from which the program may determine if more explanation of a subject is necessary or if more advanced work should be provided. The result of this is a genuine dialogue between program and student. While a certain level of student knowledge must be assumed, the emphasis in tutorial programs is more the development of understanding rather than just the revision of material already covered [5,6].

2: In drill and practice programs the computer presents a series of self-contained questions to the students and processes the answers [7]. This simple format is not suited

to the introduction of new material so drill and practice programs are limited to use as revision aids or for the testing of students. These programs are potentially more flexible than sets of written questions as they are able to give immediate feedback to the student if an incorrect answer is entered. This feedback may take the form of a hint or a detailed explanation of the correct answer. A well written drill and practice program will also include provision to alter the number or nature of the questions in a particular subject area depending on the students performance. The automatic collection and processing of results within the program is important to the teacher as it not only provides data on individual students, but it may also give overall performance statistics to aid in course evaluation. It may also indicate the suitability of individual questions.

3: A simulation program is one in which the behavior of a physical system is mimicked in some way by the program through the development of a mathematical model. Major uses for such programs are in teaching laboratories where actual experimental equipment is too complex, inconvenient or expensive to use, and in the modelling of industrial processes. Generally simulation programs are set up for use by individuals or by small groups of students so that by exploration of the effects of altering the model parameters a greater understanding of the model and hence of the actual physical system may be obtained [8].

4: As the name implies demonstration programs are used

by teachers in a classroom or lecture situation as a visual aid for the presentation of teaching material. Generally pictorial in nature the material presented via a computer is similar in nature to the material presented by other picture presentation technology such as slides, film or video. The advantage of using computer generated graphics is that much greater flexibility in the planning and presentation of lessons with a variety of presentation styles is possible than with other techniques.

5: Games in CAL refers to programs which involve a competitive element between student and computer. Such programs involve the scoring of points or the attainment of a goal or condition during the process of which knowledge of the subject being taught must be exhibited. Providing the motivation to learn, rather than presenting new material is the main reason why such games are written since the format of educational games is not normally suitable for transmitting more than a small volume of information about a subject.

6: Auto-elaborative CAL is designed to provide experiential learning situations for individual students [9]. When using a program of this type the student is allowed to explore the characteristics of the programs performance. By observing the response to program input the student should be able to discover information about the subject matter of the program. Guidance may be given in the form of goals to be achieved or problems to be solved using the program. The emphasis is on understanding the subject

since the nature of this type of program requires that the student actively seeks information rather than passively being presented with teaching material. The best known example of this type of teaching approach is the LOGO package for the teaching of a structured approach to programming [10].

2.2 EVOLUTION OF CAL

In the earliest use of computers as a teaching tool the lessons were made up only of descriptive text [11]. Such lessons were implemented on mainframe computers operating through hardcopy terminals. Initially the use of such programs was confined to "page turning" tutorial lessons for the presentation of background material. As specialized authoring languages incorporating sophisticated answer matching routines were developed, more control over the flow of the lessons could be achieved and drill and practice programs became possible. Any graphics required for lessons were either of the form of simple plots generated using the alphanumeric character set of the terminal [12] or were presented separately to the students in the form of sets of diagrams printed on cards which were referred to by the program.

The advent of readily available CRT visual display units (VDU's) as terminals initially did little to change the nature of the teaching programs since most such displays had no greater graphics capabilities than their hardcopy

predecessors. The VDU does have the advantage that text is generated more rapidly than on a hardcopy terminal but loses out in that there is no permanent record of information presented unless special provision is made for the recording and later printing of lesson dialogue. Limited special effects to enhance learning are possible on "text only" VDU's which have cursor addressing facilities [13]. However little work has been done in this field probably because the effects are not sufficiently dramatic to overcome the unpredictable nature of response times and system messages inherent in multi-user systems employed on mainframe computers.

As hardware progressed so did the sophistication of CAL software being produced [14]. However in the absence of a graphics capability on terminals this software was still restricted by the limitations of an alphanumeric output.

The high cost of graphics capable terminals prevented the widespread use of graphics in CAL until relatively low cost microcomputer systems became available. These systems were initially equipped only with low resolution or modifiable character graphics, but provision of graphics generation facilities with acceptable resolution soon followed.

With the provision of such relatively low cost, portable systems capable of high resolution graphics, software was soon written to make use of these features as well as colour and sound effects if they were also available. These new features tended to be used to

illustrate simulations or enhance tutorial programs [15], although educational games modelled on the styles adopted by the entertainment industry also became more common.

In order to make more constructive use of the software being written, attempts have been made to integrate a series of tutorial programs to follow a course of study rather than each program teaching one aspect of a subject in isolation [16,17]. Much of the work done in this project has dealt with collecting other types of teaching program in a similar manner. In this way it is hoped to increase their teaching effectiveness as well as evaluate the future potential of an integrated approach to software.

2.3 COMPARISON WITH OTHER METHODS

One of the major advantages of the computer as a piece of educational technology is its versatility. It may fill a number of different roles associated with a wide variety of teaching techniques and apparatus. In some cases the computer may replace existing equipment, while in others it may greatly enhance the teaching of a subject [18].

Tutorial programs may be used in a similar role to that of a textbook although the differences are such that they should be considered complementary rather than alternative teaching resources. Well written tutorial programs are able to guide a student through the material being presented in a more flexible manner than is possible with a book. A computer program can add or remove material from a lesson

depending on the students' responses to questions asked as well as having the potential for dynamic graphics and sound effects for emphasis. Books on the other hand can contain a great deal more information and because of their relatively low cost, easy availability and portability are able to be used in many more situations than is economically possible with current computer technology.

The use of drill and practice programs in place of written questions is a well established use of computers in education offering a number of clear advantages over the older technique. The advantages of a computer based question system are immediate correct response to student input, hints for incorrectly answered questions, automatic record keeping and variable lesson structure depending on student performance [16].

In the field of classroom or lecture theatre demonstration there are a number of ways in which pictorial information may be presented. These methods, which include overhead projectors, slide projectors, film and videotape systems, are collectively able to provide most types of desired display but no single device is able to cater for all requirements. Computer generated graphics demonstration programs offer display functions provided by most of the more conventional techniques plus some additional benefits. The features offered include randomly selectable "slide show" presentations, animated or dynamic displays, sound effects, input dependent displays including simulations and stepwise generation of displays.

2.4 THE PLACE OF CAL IN CHEMISTRY

General CAL techniques, such as tutorial or drill and practice may be applied to any subject area with little variation in style, but some techniques are especially suited to the type of material presented when teaching chemistry. These areas of particular interest to chemistry are simulations and demonstrations where graphics are involved.

A variety of complex and expensive instruments are used in chemical research or industry which cannot normally be provided in teaching laboratories for "hands-on" use. Examples of such instruments include NMR machines, mass spectrometers and X-ray diffractometers, together with smaller facilities such as infra-red spectrometers and chromatography columns. It is in such cases that computer models of instruments are of most use since a computer running the appropriate simulation program may easily be made available to students. Similar programs may also be made available to teachers as lecture demonstrations in situations where a practical demonstration of the actual instrument would, even if possible, simply show a "black box" with no details of the working of the machine.

However of even greater potential use in chemical education is the use of computer graphics to illustrate chemical concepts. As will be shown during the course of this thesis, in teaching symmetry, computer graphics may be used to illustrate concepts in ways not otherwise possible.

Symmetry is of course not the only aspect of chemistry suitable for computer graphics. Other obvious candidates for consideration include reaction mechanisms in all branches of chemistry, organic synthesis, the bases for the various types of spectroscopy, stereochemistry and chromatography. For most of these topics computer teaching programs of varying quality have already been written but the indications are clear that there is considerable scope for further development.

CHAPTER III

SYMMETRY

Symmetry in chemistry was chosen as a suitable topic for the application of computer based education techniques. It is a topic in which arrangements in space are a central concern and the power of animated graphics to convey this information gives computer methods a potential advantage over conventional techniques. The existing strong departmental interest in the field of X-ray crystallography was also a factor in the choice of subject area since many applications of symmetry in chemistry are in crystallography.

3.1 ASPECTS OF SYMMETRY CONSIDERED

Consideration was first given to the ways in which symmetry is represented. Crystallographic symmetry is based on a small number of elementary operations. These operations are identity, inversion, reflection, rotation and translation plus a small number of operations formed from the combination of two others. They may be treated mathematically as individual algebraic functions directly related to the physical nature of the operation, but they may also be collected together into "groups" of operations which conform to the rules applying to mathematical group theory [19]. Our prime concern with these groups, some of

which have particular relevance to crystallography, was not their theoretical nature, which is well understood, but rather with their physical characteristics and representation.

In the case of three dimensional point groups [20], which are used to characterize the symmetry of molecules and crystals, the standard stereographic projection diagrams are examined including the way in which they are built up from individual symmetry operations. The addition of symmetry operations incorporating translational components to point groups produces space groups which are used to characterize the arrangement of structural units within a crystal. Both two and three dimensional space groups are considered with particular emphasis on the three dimensional case where both the nature of the individual constituent symmetry operations and the way such operations are combined to form space groups may be represented by one standard diagram. The way in which unit cells may be chosen in a crystal lattice and the various forms these unit cells and lattices may take are also examined together with the relevant nomenclature.

The applied side of the field is approached through examination of the packing of atoms and molecules in unit cells and metallic lattices. The labelling of crystal lattice planes through the use of Miller indices is also treated as it provides a lead in to the examination of X-ray diffraction by lattice planes in terms of Braggs law [20]. A direct link to practical crystallography is given in the examination of the physical interpretation of the Patterson

function and the uses to which it may be put in the solution of crystal structure.

3.2 CURRENT TEACHING METHODS

The teaching of symmetry concepts in the lecture situation depends to a large degree on the use of projection devices to display static pictures. In practical exercises models of molecules or other shapes containing symmetry may be used as may various forms of diagram.

In the case of basic symmetry operations, the diagrams used to illustrate them tend to be of a "before and after" nature where an object is shown together with its transformation after it has been operated on by the symmetry operation being demonstrated. Point groups and space groups are demonstrated in the same manner but in these cases the diagrams are usually of a standard nature and several individual operations are encompassed within one diagram. The teaching of point groups is often associated with the use of molecular models since it is a relatively simple exercise, in concept, to show the individual symmetry elements in a molecule, or to ask students to identify them. Another common use of models is to demonstrate crystal packing since this is often difficult to do with a two dimensional diagram [21].

Demonstrating the physical interpretation of the Patterson function can be attempted by displaying a diagram of a unit cell and generating a corresponding Patterson

vector map by either drawing in the appropriate vectors in both diagrams or by overlaying images of the unit cell on the vector map. In either case only the simplest examples may be chosen to ensure the picture being displayed does not become too crowded. Such a technique requires either a large number of pre-prepared pictures or considerable drawing to produce the desired results, and this problem is exacerbated if Harker sections, weighted peaks and vector superposition are to be considered.

The use of computer demonstration programs to replace some or all of the projected static displays offers a number of advantages. Of particular significance is that the display need no longer be static in nature but may involve animation or other special effects to follow the path of an operation or process. This together with the facility to readily build diagrams in a stepwise fashion would appear to offer the best avenue of approach when considering ways to improve educational strategy in this subject.

CHAPTER IV

STRUCTURE DETERMINATION

In order to acquire direct experience with applications of symmetry concepts in crystallography and to provide ideas for incorporation into teaching programs, it was decided to undertake a crystal structure determination. Experience of the practical nature of crystallography was necessary in order to ensure that the approach taken in programs designed to teach theoretical aspects of symmetry in crystallography was not inconsistent with practice.

4.1 PRELIMINARY STUDIES

The determination was undertaken on well formed orange-red crystals of a compound provided by Dr D.A. House which was believed to consist of chromium(III) ions coordinated to a hexa-methyl derivative of cyclam [22] (see figure 4.1). It was initially suggested that water molecules would occupy the two octahedral coordination sites left vacant by the cyclam derivative, with three nitrate groups to act as counter ions to the coordination complex.

Preliminary X-ray diffraction studies using a precession camera indicated that the space group of the compound was either Cc or C2/c. Diffraction photographs also yielded approximate unit cell parameters which together with an estimate of the density, based on known densities of

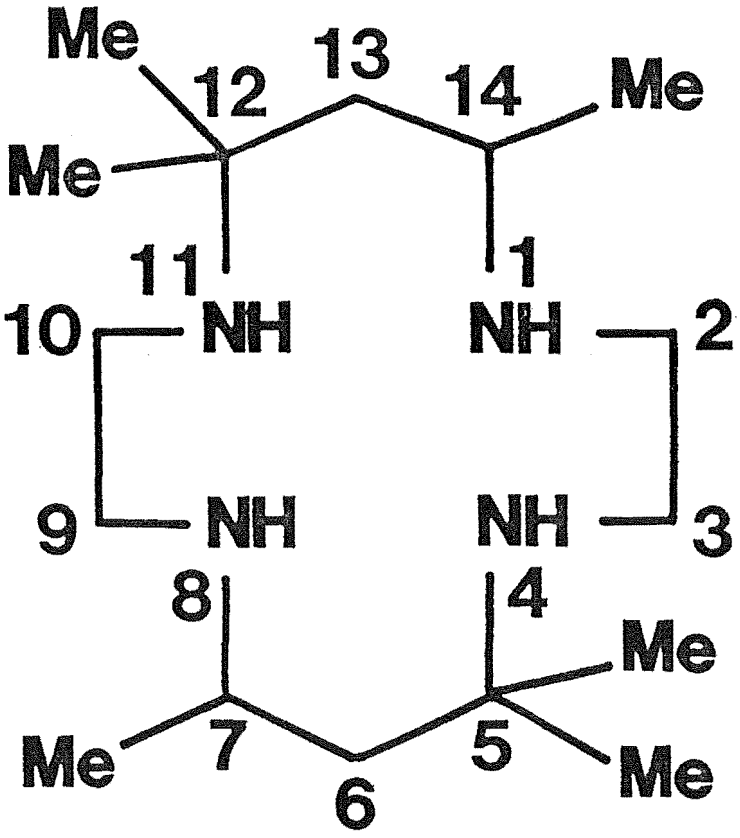


Figure 4.1 Teta (7R, 14S).

similar compounds, indicated four molecules per unit cell. This information strongly suggested that the space group was Cc. The alternative of C2/c would imply that the asymmetric unit included only half a molecule, i.e. that the molecule as a whole as well as one of the three nitrate groups possessed a two fold rotation axis or a centre of inversion. The subsequent structure analysis confirmed that the space group was Cc.

4.2 ELUCIDATION OF CRYSTAL STRUCTURE

Initial data collection took place on a Hilger and Watts four circle diffractometer. The position of the chromium was found through examination of peaks found in the Patterson map. At this stage standard "heavy atom" phasing using difference fourier techniques gave ambiguous results. Direct methods calculations were then undertaken and yielded the positions of the atoms in the coordination sphere of the chromium. Using these coordinates to aid the phasing process it was possible, using difference fourier and least squares refinement techniques, to determine the positions of all the non hydrogen atoms in the macrocyclic ligand. There remained considerable uncertainty in the positions of the atoms of the axial ligands and the nitrate groups. However it was clear that only two nitrate groups were present so one of the axial ligands was presumed to carry a negative charge to give the overall charge balance.

With only two nitrate groups in the structure it now

appeared possible that the complex could possess a centre of inversion located on the chromium thereby making space group C2/c a possibility. In order to test this possibility the charged axial ligand was initially presumed to be a hydroxide group in statistical disorder with the water molecule in the other axial position. Again refinement of the structure in the areas of the axial ligands and the nitrate groups gave unsatisfactory results even with full anisotropic refinement being used.

By examining electron density contour maps of the region around the axial ligands and with reference to the original reaction conditions for the crystal it was deduced that one of the axial ligands was a chloride ion and the space group was confirmed as Cc. However, satisfactory refinement was still not possible even though the structural information was now believed to be complete.

Data were then collected from another crystal using a Nicolet R3m diffractometer which yielded a larger and more precise data set than the first machine used. From this second collection of data which did refine in a satisfactory manner it was possible to confirm the structure eventually elucidated from the original data.

4.3 STRUCTURAL INFORMATION

The full name of the complex is trans-aquachloro [(1S,4S,7S,8R,11R,14R)-5,5,7e,12,12,14e-hexamethyl-1,4,8,11-tetraazacyclotetradecane] Chromium(III) Nitrate. The atom

labelling scheme is the same as in figure 4.2. Crystal data and experimental parameters are contained in table 4.1.

The central chromium is co-planar with the four coordinating nitrogens of the macrocycle while the water and chloride ligands occupy the axial positions. The secondary NH protons of the macrocycle are in the lowest energy RSSR (meso) configuration, and likewise the methyl groups at the 7 and 14 positions adopt the low energy equatorial positions. The overall configuration of this hexa-methyl cyclam complex is essentially the same as that of the parent chromium cyclam complex [23].

4.4 DERIVED EDUCATIONAL APPLICATIONS

Although program SGROUP (see chapter 7) had been written prior to the commencement of the structure determination, a number of changes were made to the way the molecular display was handled as a direct result of experiences gained with molecular orientations and crystal packing during the course of the determination. Less directly the problems encountered with space group identification and molecules on special positions were an influence when constructing a set of exercises to guide individual use of program SGROUP and the drill and practice programs intended for use subsequent to SGROUP (see chapter 8).

When using the Patterson function to determine the position of the chromium, the output from the program which

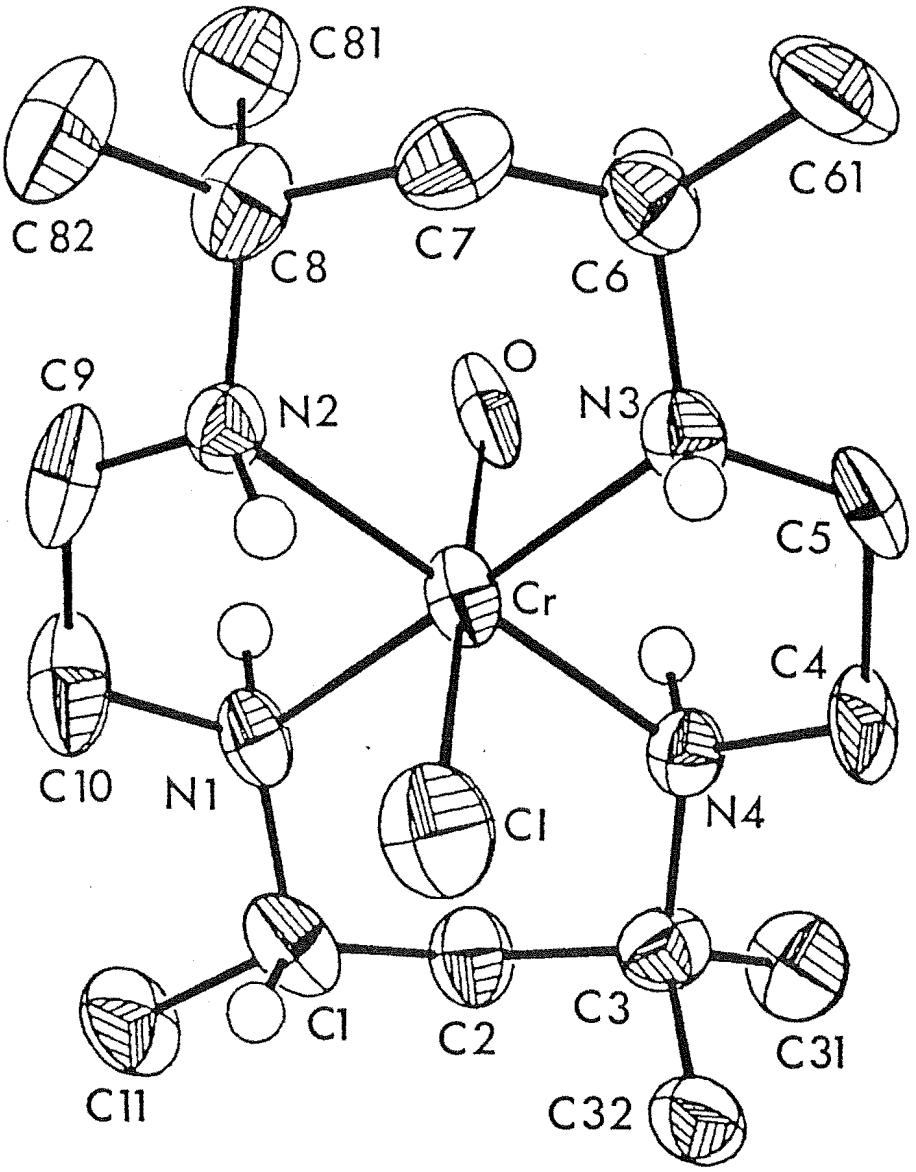


Figure 4.2 A general view of the complex cation, trans-CrCl(teta)-$(H_2O)^{2+}$.

Formula	C H N O CrCl
Formula Weight	513.96
Crystal System	Monoclinic
Space Group	Cc
a	14.560(5) Å
b	11.740(5) Å
c	14.772(7) Å
β	110.83(3)°
V	2360.0 $Å^3$
Z	4
d	1.447 Mg m^{-1} (calc)
F(000)	1092
Crystal Dimensions	0.25 x 0.21 x 0.16 mm
μ(Mo Kα)	0.632 mm^{-1}
T	293 K
Radiation	Mo (Kα) (λ=0.71069 Å)
Total reflections	2367
Reflections used in refinement	1894 for which I>3σ(I)
Number of structural paramaters refined	333

Table 4.1 Crystal data and experimental parameters for $[CrCl(H_2O)(C_{16}H_{36}N_4)](NO_3)_2$

performed the calculations consisted of a list of coordinates corresponding to the highest peaks in the patterson map. Except for experienced crystallographers the interpretation of such a list may be difficult. A teaching device to convey the physical interpretation of the Patterson function was therefore seen to be desirable.

The other aspect of a structure determination which appeared in need of teaching resources and was suitable for CAL applications was the area of basic theory. The programs which eventuated dealt with Braggs law and Miller indices, both of which are subjects covered in undergraduate courses and hence these programs have considerable practical use (see chapter 11).

CHAPTER V

THE COMPUTER

For educational software to have maximum effect it should be available to the greatest possible number of potential users. Assuming a method of distributing software is available, the most significant limitation in its use is the number of potential users who have a compatible computer system. Naturally any computer system chosen must also have the hardware and software capabilities required to develop the envisaged software. Other factors such as cost effectiveness, the number of other users of the system, the availability of utility software and service backup are also important when choosing a computer system.

5.1 REQUIREMENTS

Since this project places emphasis on the use of computer graphics, the first requirement of the chosen computer system is that it has an adequate graphics capability. The graphics system used should be self contained and should include provision to manipulate an individual unit of the screen display independent of any other unit. These units, called pixels, may or may not be constrained in groups to make up individual characters. An approximate guideline for the generation of a sufficiently precise display is that there should be a minimum of 250

pixels across the screen by 150 pixels down [24]. The provision to control the colour of the pixels is a desirable but not essential feature which adds to the effect of the graphics display [25], although at the cost of a greater computer memory requirement.

To make the most effective use of the available graphics hardware, a computer must have graphics generating software which is both convenient to use and flexible. To write educational software, languages must be available on the computer which fill the particular needs of this field while still retaining the graphics capabilities. The amount of computer memory and peripheral storage capacity must not be so small as to be unduly restrictive on program size and the computer must have adequate processing speed.

Hardware portability is another important consideration, bearing in mind that the computer may be used in a variety of different physical situations including lecture rooms and teaching laboratories. The ability of the computer to drive more than one display simultaneously is also important in this context as a large room may require multiple displays. Peripheral devices other than display screens may also be required so the support of interface devices is a desirable feature.

5.2 THE CHOICE

The first decision to be made was whether programs should be designed for a mainframe computer via a graphics

capable terminal, or for a microcomputer. The decision to use a microcomputer was made on the grounds that a microcomputer could provide all the features necessary for writing graphics based CAL programs at a lower cost and with a greater potential number of users than a mainframe with a graphics terminal. A mainframe based teaching system would also lack the portability of a microcomputer and would also be subject to access delay problems associated with a multi-user system.

An Apple][+ microcomputer was available and so was examined for possible use in comparison with other microcomputers available at the time the project was started. The Apple had been initially procured because of its superior graphics capabilities when compared to its competitors, which, at the time, included the Commodore PET, the TRS 80 model 1, and the Northstar Horizon [26]. The Apple][computer has individually definable pixels in graphics mode with a resolution of 280 pixels across by 192 pixels down the screen which meets the criteria for an adequate graphics display. None of the other machines possessed superior graphics resolution and control nor did they offer the colour facility that the Apple offered. Thus if the Apple could meet the other eligibility criteria it would be the most suitable machine to use.

The Apple][+ computer could accommodate 64 Kilobytes of RAM storage (i.e. 64K) with a further 12K of ROM containing the operating system and the Applesoft language. This amount of memory is sufficient for the development of

most educational programs provided adequate peripheral disk storage is available. Up to six 5 1/4 inch floppy disk drives may be added to the Apple, each capable of providing quick access to 140K data on a single disk, while hard disk systems available range from 5 to 20 Megabytes storage capacity. The Apple also possessed the greatest range of languages and graphics utility software of any of the contending machines thereby making it the most useful for education program development. The utility of the Apple was also enhanced by the readily available advice about use of the Apple from user groups, magazines, dealers and other Apple users.

The Apple is sufficiently portable to be moved easily to lecture theatres or laboratories without difficulty, and without the need to provide special facilities. A carry case and a trolley were devised to make any transportation operations more convenient although such devices are not essential. Several video monitors may be driven simultaneously from the Apple video output for use in large lecture rooms or in other situations where more than one display is required.

A number of the more recently available microcomputers, for example the IBM PC, the BBC Micro, the Commodore 64, and the Apple Macintosh, have hardware characteristics which match or exceed the performance of the Apple][+ and its successors. However because the Apple][series is very widely used in the chemical education field and in chemical applications generally, and still has a greater range of

available software than the competition, it remains a suitable machine for writing educational programs. The improved hardware performance of the later models in the Apple][series ensures that this computer will continue to be an adequate machine on which to develop educational software in the immediate future.

5.3 COMPUTER CONFIGURATION

When deciding the configuration of the computers to be used in this project it is important to consider which features and peripherals are most likely to be available to intended users of the software to be developed. Any software which requires the presence of non standard hardware features was to be avoided. However it is possible to write software, which while making use of non standard features, may also be executed on a computer lacking the additional features. Hardware modifications or additions to aid the development of programs may be used without making the execution of the programs dependent on the extra hardware.

To determine the minimum hardware configuration necessary for program execution, the language system to be used for writing the programs first had to be considered. Since all programs developed were to be executed under the Pascal operating system (see chapter 6), the minimum memory required on the computer was the 64K required by Pascal. Programs to be executed under the Pascal system cannot be

loaded via the cassette interface so a suitable disk drive system must be present for the loading of the operating system, programs and data files. The eventual specification for the minimum configuration was an Apple][series computer equipped with 64K memory, a monochrome video monitor and one standard Apple 5 1/4 inch floppy disk drive. During the course of this project programs were written which were capable of using a color monitor, lower case display, a second disk drive or a printer, but all these programs would still execute on the specified minimum configuration.

Initially the computers used for developing programs were Apple][+'s with the specified minimum configuration plus a second disk drive, essential to the development of large Pascal programs. Lower case display chips were added to enhance the readability of the text display, while a colour monitor was procured so that colour capable software could be tested.

Certain devices were used during program development but were not required for execution of any of the programs. These included a clock device for the dating of files and the possible timing of lessons and a 128K memory card, acting as a pseudo disk drive to give faster program compilation.

The Apple][+ computers were eventually replaced by the Apple //e which incorporates 64K memory and a lower case display as standard features. The Apple //e also features an enhanced keyboard and the possibility of increased

resolution of the graphics display. The computers obtained were the "Euroapple" model which has a PAL colour video output included as a standard feature. The computers were supplied with expansion cards which generate an 80 column text display rather than the standard 40 column display. While possessing a number of new features the Apple //e was almost completely software compatible with the Apple][+, which meant that software developed on the Apple //e could be executed on the older model machines.

5.4 USE OF PRINTERS

A printer attached to a computer system is of use in most phases of program and documentation development, as well as during actual teaching sessions. Printer listings of program source files are an essential aid in the detection of program bugs as well as providing hardcopy backups in the event of accidental erasure or destruction of the disk files. Some printers are capable of generating hard copy images of the computer graphics display and therefore are of great value in development of programs using graphics. In a teaching situation, hard copy of data and graphics images from a program may be provided to students so that a permanent record is available to them for revision purposes.

The principal features which must be considered when choosing a printer for use in education are speed and quality of the print, graphics capabilities, noise level

during operation and mechanical durability. Other useful features are friction feed for single sheets of paper, variable print fonts, user definable print fonts, subscript and superscript capability and a print buffer. Some of these features are related to the interface between computer and printer so the choice of printer also involves finding the most suitable interface. The final choice, as with the choice of computer, will greatly depend on cost and availability of service, but unlike the choice of computer little emphasis need be placed on the number of other users of the same type of printer. The printer finally chosen was a C. Itoh 8510 which is essentially the same as the Apple DMP printer marketed with the Apple computer. It is driven through a parallel interface containing firmware for a bit image graphics dump and is relatively quiet, fast and robust.

CHAPTER VI

CHOICE OF LANGUAGE

Once the decision to use the Apple][computer had been made, a further decision had to be made regarding the operating system and programming language(s). The choices of operating system and language are interdependent as a particular language is usually only available with a specific operating system. The object therefore was to find a language suitable for the writing of the desired programs, working under the most convenient operating system available [27]. An additional consideration was that the most suitable language for lecture demonstration, and simulation style programs may not be the most suitable for interactive tutorial programs. If more than one language was to be used to overcome this problem it was desirable that they all run under the same operating system.

6.1 THE CHOICE: PROGRAMMING LANGUAGES

The first system to be considered was the Apple][operating system which is installed in ROM on the Apple][+ and Apple //e computers. This system incorporates an implementation of the BASIC programming language known as "Applesoft" [28]. A disk operating system, Apple DOS 3.3 is designed specifically to be used with it.

The major attraction of the Apple operating system and

DOS 3.3 was that it was the most commonly used system available on the Apple with the greatest amount of available documentation and most utility programs. Applesoft BASIC incorporates commands to make use of the Apple's high resolution graphics capabilities, in the form of line and shape drawing commands on a cartesian grid system. Since the language is based on ROM, Applesoft has a low memory requirement compared to other systems, although the organization of the Apple memory does not allow full utilization of the available space if graphics are to be used.

However both the Applesoft language and its operating system have a number of serious disadvantages. Applesoft like most implementations of BASIC is a line based language with no block structuring provisions. This greatly reduces the ease with which large or complicated programs may be developed. Since only the first two letters of an identifier are significant, meaningful identifiers cannot easily be used and the programs as a consequence are hard to read and difficult to debug. The tracking of errors is further inhibited by the absence of descriptive error handling.

However once a syntax error has been found and corrected there is no need to recompile the program since Applesoft is an interpretive language, i.e. the program is executed directly from the source program rather than from a compiled code file. This is a convenience in correcting mistakes but at the expense of greatly reduced execution

speed. For larger programs the slow execution can severely restrict their general usefulness.

Models of the Apple before the //e had no provision for lower case on the text screen and applesoft had no provision for adding text to any place on the graphics display although utility packages became available which provided both of these features [29]. Inclusion of such utilities reduced the amount of available memory for programs, but did not overcome the major problems of slow execution speed and lack of structured programming features. Subsequent Applesoft BASIC compilers did little to alleviate the overall shortcomings of the language. The advantages of an interpretive language were lost and the effective increase in speed was only small since the compilers have little effect on graphics commands which make up the bulk of most teaching programs. Compilers also tend not to be compatible with most of the available utility packages.

The CP/M operating system for the Apple computer has several in-built utility features not available in the standard Apple operating system [30]. The CP/M system is packaged with two versions of the microsoft BASIC language. The two versions differ only in that one of them (MBASIC) does not have the high resolution capabilities of the other (GBASIC) [31]. Both BASIC's are very similar to Applesoft but include several minor features not available in Applesoft.

Although slightly superior to Applesoft, GBASIC retains the disadvantages of a non structured, interpretive

language. GBASIC has the additional disadvantage of a large memory overhead. The CP/M operating system is superior to the native Apple system and Apple DOS 3.3, but again the memory overheads are greater, as is the amount of disk space required. The other major drawback of the CP/M system is the need for a Z80 coprocessor card. Since this card is not a standard feature on Apple two series computers, the use of any software written to use the CP/M system will be greatly restricted. Many other computers, other than Apple, use the CP/M system but they do not use the same graphics system so the transfer of programs to other systems is not feasible.

The system eventually chosen for programming applications was the Apple Pascal system [32]. Pascal is a highly structured and flexible language which facilitates the writing of large or complex programs [33,34]. The graphics command set is an adaptation of turtlegraphics [10] and is more powerful than that contained in Applesoft. Pascal is a compiler based language giving it a faster execution rate than interpreted forms of BASIC. The compiler has more flexible options and better error handling than any BASIC compiler currently available.

The Pascal operating system contains a powerful and easily used text editor and file handling utility, together with an assembler, a file linker and a number of utility programs [35]. These features make it significantly superior to the other operating systems considered.

The Pascal system does possess some of the disadvantages noted in relation to the CP/M system. Pascal

has large memory and disk space overheads relative to Applesoft, and while it does not require a Z80 card, it requires an Apple equipped with 64k ram to function. As well a second disk drive is highly desirable during program development although not for program execution. However the superior nature of the Pascal language for program development, and the power and ease of use of the operating system more than compensate for such drawbacks [36]. A more complete description of the Pascal system is given in section 6.5.

6.2 THE CHOICE: TUTORIAL LANGUAGES

Although suitable for most programming applications Pascal, like most general purpose languages, is not well suited to interactive tutorial or drill and practice programs. In most cases a specialized tutorial language is more suitable (see section 10.2).

The PILOT (Programmed Inquiry, Learning Or Teaching) language was originally developed in 1968 at the University of California, San francisco [37]. Implementations of COMMON PILOT, a version of the original language developed at Western Washington University, constituted the most commonly used tutorial authoring languages used on small computer systems at the time this project was commenced. "Apple PILOT" the most powerful of the available implementations of the language was considered for use [38].

Apple PILOT contains powerful routines for the display

of graphics and the generation of sound effects, as well as the standard PILOT features for the display of text and the processing of input [39,40]. The operating system is based on the Apple Pascal operating system making it possible to compile and execute pascal programs under the PILOT operating system, but with memory restrictions which limit the size of the programs severely. PILOT programs may not be executed under the Pascal operating system.

Although it initially appeared to be suitable for writing tutorial programs a number of deficiencies appeared when PILOT was used to write the first two sections of the SGROUP questions tutorial (see chapter 8). Precise formatting of the text part of the screen display proved difficult as the automatic formatting feature of PILOT could not be disabled and checking of the final format was not possible while in the text editor. The files containing graphics screen images to be displayed during the course of a program took up a great deal of disk space, as did the operating system, and hence there was a severe restriction on the number of programs which could be placed on one disk. The other graphics features incorporated in the language were too slow for the generation of all but the most simple images.

The problem that eventually led to the search for another language was the way in which PILOT programs were broken into arbitrary sections by the program interpreter because of memory restrictions. This led to unacceptable breaks during the execution of the program caused by the

need to load a section of the program from disk.

An implementation of the STAF (Science Teachers' Authoring Facility) system, developed at Leeds University [41,42], was in use on the mainframe computer system (Burroughs 6900) at the University of Canterbury for the presentation of interactive tutorials. Full implementations of the STAF system on the Apple computer proved to be too slow in execution to be of use, especially when the lack of graphics facilities was considered. A graphics oriented tutorial language, based on the STAF language, specifically designed for the Apple computer was developed by R. D. Draper at the University of Canterbury. While this system, named MASTER (Microcomputer Adapted Staf for Teaching and Educational Research), was not as flexible as PILOT, it contained all the features necessary to create the intended tutorial Programs. Since the author of the system was available for consultation, any deficiencies or defects in the system could be easily corrected by modification of the code for the MASTER interpreter.

The MASTER system offers a number of advantages over PILOT including more compact disk file formats, complete Pascal compatibility, and complete control over the format of text on screen. Since any MASTER lesson is entirely loaded into memory there are no breaks required during execution to load information from disk. A convenient method of processing student responses, including automatic storage of all responses, was another reason for MASTER to be eventually preferred as the most suitable language for

writing tutorial programs.

6.3 OTHER LANGUAGES

A number of other operating systems and languages have become available on the Apple since the beginning of this project. Although none of these systems possesses features sufficiently advanced to replace Pascal or MASTER, a number of them do have features of interest to those wishing to write CAL programs.

PRODOS is designed to replace Apple DOS 3.3 as the disk operating system for use with Applesoft BASIC [43]. PRODOS incorporates a number features which overcome the deficiencies of Applesoft. PRODOS does not however change the Applesoft language or the program editing facilities and so does not provide a desirable alternative to Pascal.

Apple Fortran is an implementation of Fortran 77, and runs under the Apple Pascal operating system [44]. Fortran has been very popular as a scientific programming language on mainframe computers. A large number of programs relevant to CAL have been written in Fortran and the Apple version of the language may be used to implement these programs. The code files generated by the Apple Fortran system are compatible with those generated by Apple Pascal making it possible to link programs of the two languages together for greatest effect. However Fortran could not be considered as a first choice programming language since, like BASIC, it is not a block structured language.

The C programming language is a modern structured language similar in nature to Pascal [45]. The C language is widely used on minicomputer systems in conjunction with the UNIX operating system. Although C is in itself powerful and flexible the Apple implementations do not fully exploit its features [46]. The lack of standard graphics driving routines, the need for a Z80 card and the non implementation of the UNIX operating system are the major reasons for rejecting C as a language for writing programs for teaching.

LOGO is a graphics oriented language specifically designed for use as an educational tool [47]. Designed more as an auto-elaborative aid for the teaching of structured programming, rather than for actually writing teaching programs, LOGO has however been used successfully as a serious programming language by educators. LOGO contains essentially the same graphics system as Apple Pascal, and a similar structured nature, but does not have a sophisticated operating system comparable to the Pascal operating system. LOGO also has non-graphics oriented features [48], but not to the same extent as Pascal, and so does not have the same scope for utility programs that Pascal has.

GRAFORTH is an implementation of the FORTH programming language specifically designed for applications using graphics [49]. It offers the fastest execution speed and the greatest range of intrinsic graphics commands of any language tested on the Apple. The lack of floating point arithmetic and some other non-graphics intrinsic utilities, reduced screen resolution and a relatively primitive

operating system constitute the major drawbacks of this language.

Of the tutorial languages examined only one, apart from PILOT and MASTER, had sufficient graphics facilities to merit further consideration. The LESTER (LESson presenTER) system is a Pascal based language with graphics capabilities comparable to those of Apple PILOT [50]. Text formatting and answer storage are handled better in LESTER than in any other system considered. Large internal memory and disk storage requirements, and poor answer entry and matching facilities are the factors which eventually ruled against the use of LESTER.

6.4 PASCAL

"Apple Pascal" is an implementation of the UCSD Pascal system developed at the Institute for Information Science at the University of California at San Diego [51]. Apple Pascal contains a number of extensions to make greater use of the capabilities of the Apple computer. The most significant of these extensions are the implementation of the Turtlegraphics system for generating graphics, routines for using analogue input devices such as games paddles or a joystick, and routines for using the sound generating facilities of the Apple.

Since the programs written during this project require an implementation of UCSD Pascal with only a turtlegraphics extension, the only impediment to transfer from the Apple

computer to another computer running such a system is the resolution of the graphics display on the target system. Although the conversion of commands to fit a graphics display with a different resolution would be time consuming, it would not require major changes to the logic of any of the programs written for this project. Therefore the transfer of programs between systems is feasible.

Transfer of programs to computers using implementations of Pascal other than UCSD Pascal are more difficult. As well as different graphics systems, most implementations of standard Pascal have a different operating system and may lack many of the extra features incorporated in UCSD Pascal which would result in the need to write routines to replace them. Consequently the time required to implement the programs would be greater than for a UCSD system.

6.5 THE MASTER SYSTEM

The MASTER tutorial system consists of a preprocessor program and two forms of an interpreter program, all running under the Apple Pascal operating system. While the programs making up the MASTER system are written in Pascal and MASTER programs are entered via the Pascal text editor, the MASTER language itself has little in common with Pascal. Like STAF, on which it is based, MASTER is a non structured, node oriented language, with special provision for processing input strings and displaying text and graphics as they are required for tutorial lessons. Few other features

associated with general purpose programming languages are retained in MASTER since they are of little or no benefit in the writing of tutorial programs.

Once a MASTER lesson has been formulated and entered into the computer using the Pascal text editor, it must be run through the preprocessor program before it may be executed. The preprocessor program, named STAFANAL, breaks the source text file into three new files containing the string, node and operator information used by the interpreter. There are two versions of the interpreter program so that MASTER programs may be executed from Pascal command mode (program TEACHER) for examination and debugging, or executed from within another program, such as a menu program, as is needed when packages are constructed for student use (program INTERP).

A number of utility programs designed specifically for the MASTER language have also been developed. Program RECOVER is used to retrieve and display the student response files, program CHED is used to create special characters, and program PICCY is used to manipulate and store compressed picture files. A more detailed description of the utility programs is contained in chapter 13.

CHAPTER VII

THE SPACE GROUP PROGRAM

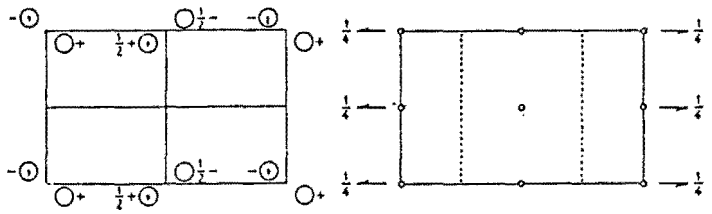
Program SGROUP was the first major program written during the course of this project for the teaching of symmetry concepts [52]. The essential function of the program is to generate a pictorial representation, in projection, of monoclinic and orthorhombic space groups, using a format consistent with the one adopted by International tables for X-ray crystallography (hereafter abbreviated to I.T.) [53] (see figure 7.1). A space group diagram is built up in a stepwise fashion by the addition of one symmetry operation at a time, under the control of the user.

7.1 PROGRAM DESCRIPTION

When the space group generation is complete the normal screen display consists of a rectangular unit cell outline, of standard size, together with the combined contents of the equivalent position diagram and the symmetry elements diagram from I.T. using standard I.T. symbolism.

The display is built up one operation at a time by keying in, from a displayed list, the three letter code corresponding to the first three letters of the name of the operation. For example REF is the code for reflection. The parameters required for any operation are then entered in

Monoclinic $2/m$ $P\,1\,2_1/c\,1$ No. 14 $P\,2_1/c$ C_{2h}^5



Origin at $\bar{1}$; unique axis b 2ND SETTING

Number of positions, Wyckoff notation, and point symmetry			Co-ordinates of equivalent positions	Conditions limiting possible reflections
				General:
4	e	1	x,y,z; $\bar{x},\bar{y},\bar{z}$; $\bar{x},\frac{1}{2}+y,\frac{1}{2}-z$; $x,\frac{1}{2}-y,\frac{1}{2}+z$.	hkl: No conditions $h0l$: $l=2n$ $0k0$: $k=2n$
				Special: as above, plus
2	d	$\bar{1}$	$\frac{1}{2},0,\frac{1}{2}$; $\frac{1}{2},\frac{1}{2},0$.	hkl: $k+l=2n$
2	c	$\bar{1}$	$0,0,\frac{1}{2}$; $0,\frac{1}{2},0$.	hkl: $k+l=2n$
2	b	$\bar{1}$	$\frac{1}{2},0,0$; $\frac{1}{2},\frac{1}{2},\frac{1}{2}$.	hkl: $k+l=2n$
2	a	$\bar{1}$	$0,0,0$; $0,\frac{1}{2},\frac{1}{2}$.	hkl: $k+l=2n$

Symmetry of special projections

(001) pgm; $a'=a$, $b'=b$ (100) pgg; $b'=b$, $c'=c$ (010) $p2$; $c'=c/2$, $a'=a$

Figure 7.1 Standard diagram and information for space group $P2_1/c$ from International Tables for X-ray crystallography.

response to the appropriate prompt. A complete list of the operations is given in table 7.1.

To make the program more effective for teaching a number of features not found in I.T. have been added. Numerical labels may be displayed above the equivalent position circles to aid description or keep track of positions when the view is altered (see figure 7.2). The coordinates of the equivalent position circle may be varied, e.g. to show the effect of placing an object on a special position (see figure 7.3).

The space group diagram may be viewed as a projection down any desired crystallographic axis by use of the appropriate program commands. These commands may be used at any stage during the program, but they do not regenerate the symmetry operation symbols (see figure 7.4). Thus, while the orientation of a generated space group diagram may be changed by the appropriate command, the symbols will no longer be displayed. However numeric labels are retained so that it is possible to keep track of particular equivalent positions after reorientation.

7.2 REPRESENTATION OF MOLECULES

The most important extension to the I.T. representation is the provision of a "molecule" option to replace the equivalent position circles. A ball and stick diagram of a molecule or molecular fragment is displayed instead of the circle and this fragment corresponds to the asymmetric unit

OPERATION MODE COMMANDS

code		action	required parameters
REF		mirror plane	(i) - axis perpendicular to plane (ii) - coordinate of plane (1)
GLI		glide plane	(i) - axis perpendicular to plane (ii) - direction of translation (iii) - coordinate of plane (1)
INV		centre of inversion	- coordinates of centre (3)
SCR		two fold screw	(i) - axis (ii) - coordinates of axis (2)
ROT		two fold rotation	(i) - axis (ii) - coordinates of axis (2)
CEN		lattice centring	- A,B,C or I (F is a combination of A,B,C)
CELL	*+	unit cell edges	- lengths of edges (3)
AXES	*	display the orientation of the axes	
OPN	*	reverses the operations option. (i.e. stops/starts the symmetry operation symbols being displayed)	
NUM	*$	turns on/off numeric labels above the equivalent position circles	
X-Y	*	exchanges x and y coordinates	
XYZ	*	rotates x, y and z coordinates	
REP	*+	repeats the unit cell so as to display molecules in adjacent unit cells	
DEL	*	deletes the last set of positions or molecules generated	
DUMP	*	printer dump of screen contents	
STAT		display current operational status	
MENU		display list of operations	
END		exit from operation mode	

* These command codes do not appear on the prompt line but can be used at any time in operation mode.
\+ These commands only work in molecule mode.
$ This command does not work in molecule mode.

Table 7.1 Summary of operation mode commands, as appears in the documentation for program SGROUP.

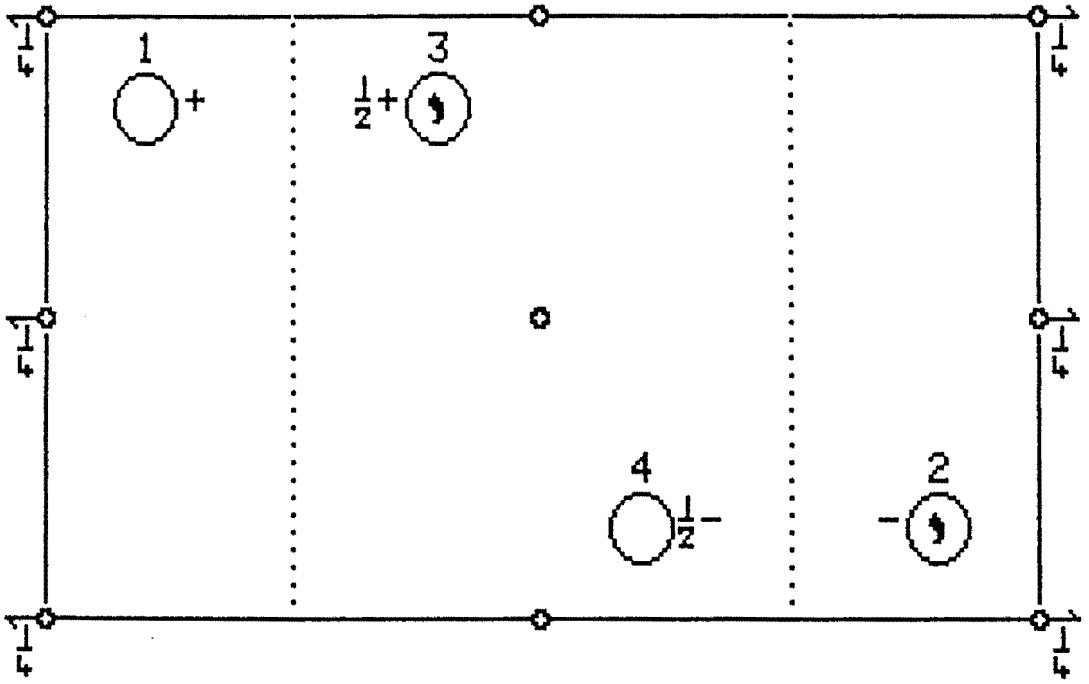
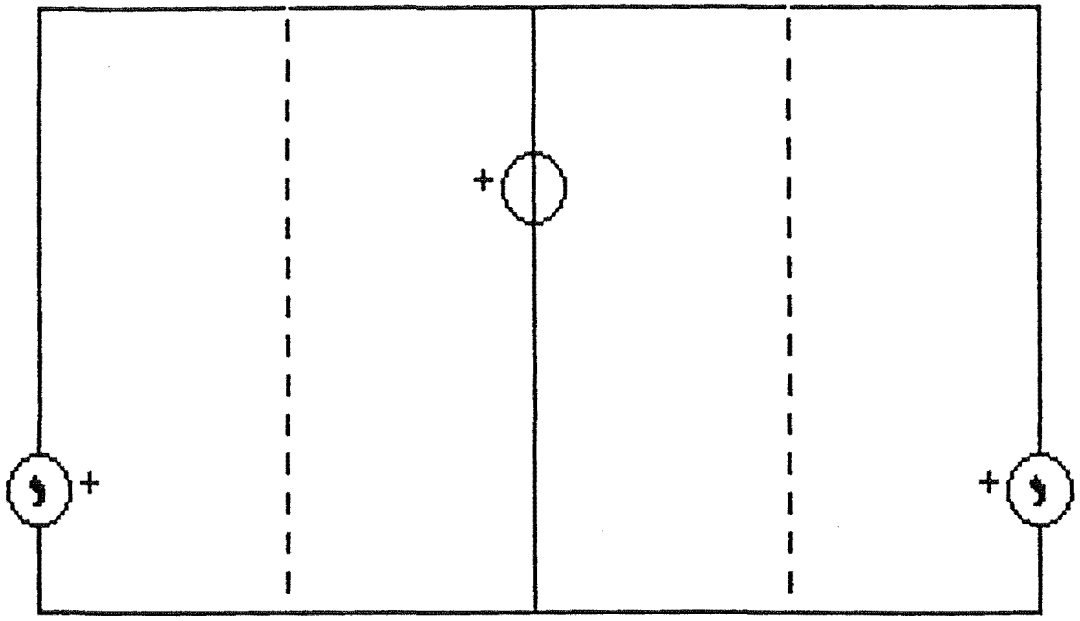
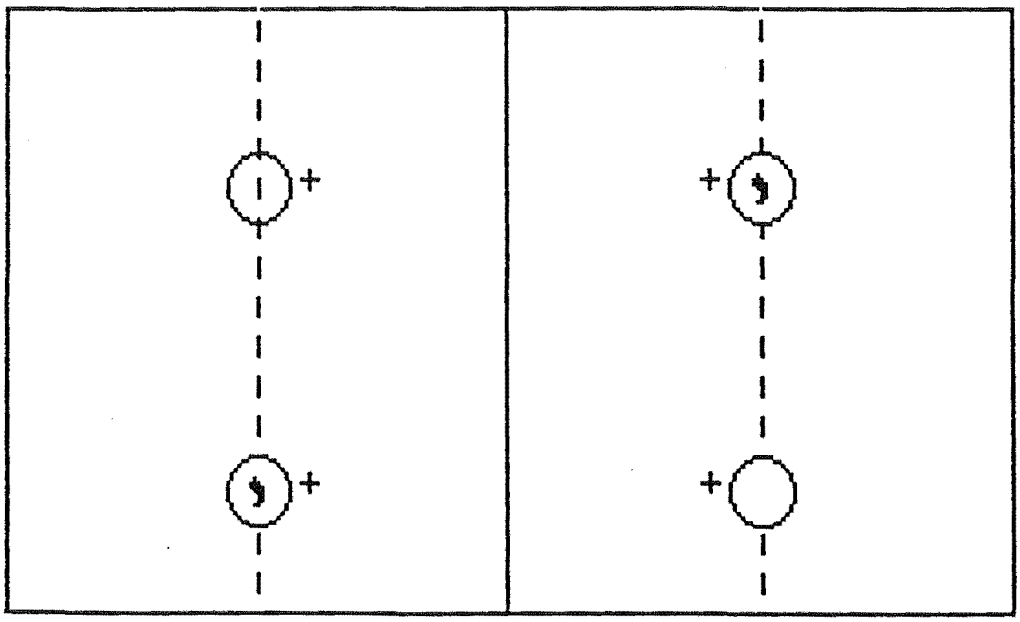


Figure 7.2 Diagram of space group $P2_1/c$ as generated by program SGROUP. This diagram includes operation symbols and numbering of equivalent positions.



(a)



(b)

Figure 7.3 Diagrams of space group Cm generated by program SGROUP. a: The circle is placed on a mirror plane which constitutes a special position. b: The circle is placed on a glide plane illustrating that this is not a special position.

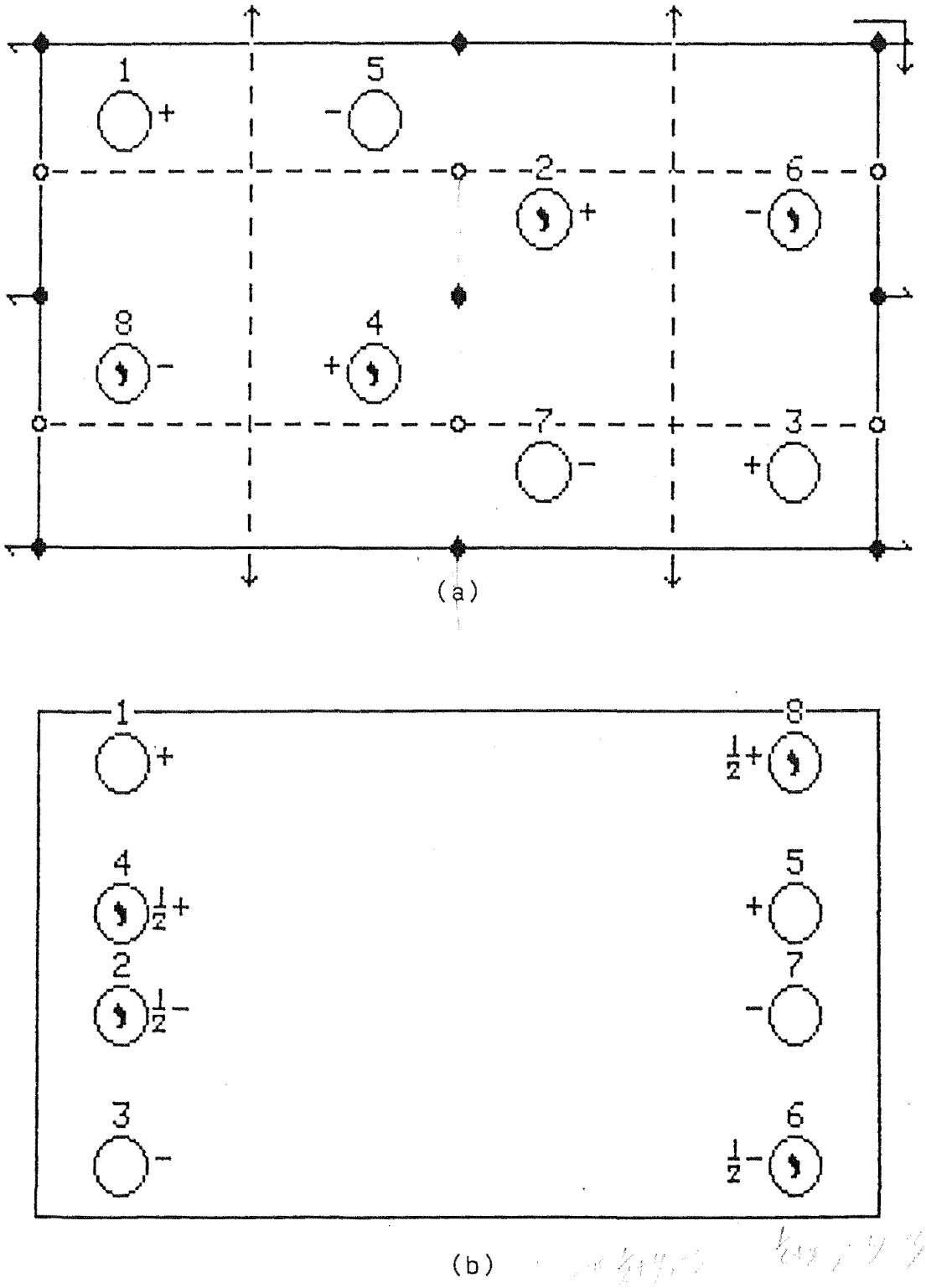


Figure 7.4 Diagram of space group Pbaa. a: Conventional view, with the a axis down the page, b across and c out. b: After use of the XYZ command. The b axis is now down the page, c across and a out.

of the crystal structure.

The information for a molecule may be either entered from the keyboard or loaded from a previously prepared file on disk, after which the symmetry operations may be added to generate a representation of the unit cell. In this mode it is important for the cell projections to be displayed with true relative axial lengths. Unit cell dimensions may be input to achieve this but symmetry operation symbols may not then be displayed.

The molecule information, contained within the disk files, is restricted to the fractional coordinates of the atoms and their connectivity. Shortage of program memory precludes additional information on atom type. For the same reason there is no provision to place labels on the fragments when they are displayed. A list of disk files used with the program is contained in table 7.2.

7.3 PROGRAM EXECUTION

SGROUP was written in Apple-UCSD-Pascal 1.1, and is executable under the Apple language system [35]. To ensure maximum transportability between systems the program is designed to run using the minimum hardware configuration described in chapter 5. Provision is made however for a printer with bit image graphics dump capabilities, and a second disk drive which may contain molecule data files. The program will execute using the standard Pascal operating system files or the modified files supplied with the

Molecule files for SGROUP.

Filename	Space group	a	b	c	Compound Name
BENZAMIDE	$P2_1/c$	5.6	5.0	21.9	Benzamide
CUACAC	$P2_1/n$	10.3	4.7	11.4	Copper bis - acetylacetonate
BRBENZCN	Cm	9.5	8.6	4.2	4-bromo benzonitrile
METHIONINE	$P2_12_12_1$	7.1	24.6	5.3	L-methionine hydrochloride
IOTHCL	Pbca	9.5	17.6	7.7	2-imino 4-oxo 1,3 thiazoidine hydrochloride
NITROMIN	$P2_12_12_1$	11.8	11.9	6.8	methyl bis-(beta chloroethyl) amine N-oxide hydrochloride (Trivial name nitromin)
DAATT	Pbaa	5.1	12.7	8.8	diacetamide (trans trans)
DAACT	$P2_1/n$	4.2	17.0	7.7	diacetamide (cis trans)
DISELENANE	$P2_1/c$	4.4	10.9	12.8	1-oxa 3,5 diselenane
FBA	$P2_1/c$	6.6	3.8	24.8	o-fluorobenzoic acid
BTM	$P2_1/c$	8.2	11.5	10.3	bis-(2 thienyl) methane
BAA	$P2_1/c$	12.6	4.3	8.0	bromoacetic acid (1)
BAA2	Pccn	23.6	6.9	5.0	bromoacetic acid (2)
MITCH	$P2_1/n$	8.0	11.2	9.9	S-methyl isothiocarbonohydrazine hydroiodide

Table 7.2 Molecule files used in conjunction with program SGROUP.

symmetry operations package. (see Appendix A)

Two utility programs have been written for use in conjunction with SGROUP. Program MOLEDIT is used to create and modify the molecule data files used in SGROUP, while PACKER converts molecule files created using the Pascal system text editor into the more compact form used by MOLEDIT. SGROUP will read both the text editor format files and the compact format files.

Documentation for program SGROUP is contained on a disk file on the symmetry operations package documentation disk. The program documentation details program execution, initialization and the use of all commands. A summary of the commands is also provided for easy reference (see table 7.1). Documentation is also provided for programs MOLEDIT and PACKER in a separate file in the same package.

7.4 USE OF SGROUP IN LECTURES

The ability to generate a space group diagram in a stepwise fashion represents the greatest advantage for a teacher using SGROUP over more traditional methods of teaching space groups, in a lecture situation. Generation of a space group diagram one operation at a time allows the teacher to draw attention to the relationship between the operation and the effect it has. The student can see the operation input and the new positions being generated. Flashing of the new positions helps to draw student attention to them.

Because of the generalized nature of the program, a teacher may use SGROUP to generate part or all of seventy different conventional space group diagrams in any of three orientations. As well, non-standard diagrams may be generated, including those with the circles used to denote equivalent positions placed on special positions or on other positions of interest such as glide planes or screw axes.

Use of molecules instead of equivalent position circles permits a wide range of teaching possibilities. A display using molecules may be used to introduce the relationship between the space group and the arrangement of molecules within the unit cell of an actual crystal structure. The structure can be shown to be built up of asymmetric units which correspond to the equivalent positions in the standard diagram. Diagrams can be generated using structures in which the molecules possess crystallographic symmetry, including situations where atoms lie on special positions [54], to show the effect of these positions in a real situation, and to show how the asymmetric unit need not be a whole molecule. The way in which molecules pack together, in the crystal lattice, may also be demonstrated using this program.

The display modifying options available in SGROUP enable the teacher to have more control over the appearance of the display. Changing the view is of most use in "molecule mode" in order to display the view with the least overlap.

7.5 DETAILS OF USE IN LECTURES

The use of program SGROUP as a lecture tool at the University of Canterbury has been mainly confined to the BSc honours part II class (3rd year).

All lectures in which SGROUP was used were conducted in a lecture theatre equipped with video monitors which could be linked to the computer used by the teacher. The monitors were set up to ensure that all students and the teacher could have a clear view of at least one of the screens.

Initially students were shown the format of I.T. and the space group used as an example was then generated using SGROUP. The generation was done step by step with explanations of the operations as they were entered and displayed. The space group usually chosen for the first example is $P2_1/c$, the most commonly found space group in crystallographic studies, with the added advantage of being a relatively simple introduction to the subject area (see figure 7.5). During the early stages of the teaching of space groups the emphasis is on the interpretation of the various components of the diagram. The significance of the equivalent positions, the symbols of the various operations and the effects of these operations are most closely examined.

Space group Pnma is also useful as an example with its greater variety of different symmetry operations (see figure 7.6). This space group is especially useful in demonstrating the concept of special positions. The

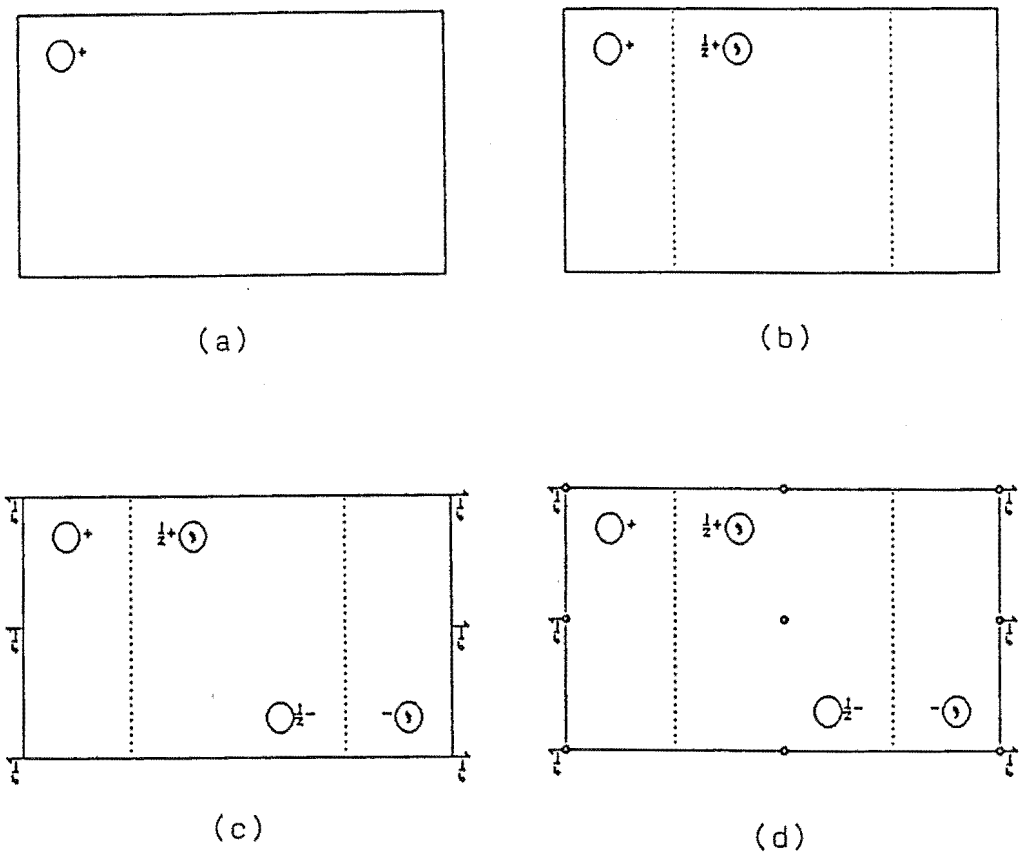


Figure 7.5 Steps during the building of an equivalent position diagram for space group $P2_1/c$. a: The initial display. b: The GLI operation is used to generate a c-glide normal to the b axis. c: A screw axis parallel to the b axis is then added. d: The implied inversion operation is added to complete the diagram.

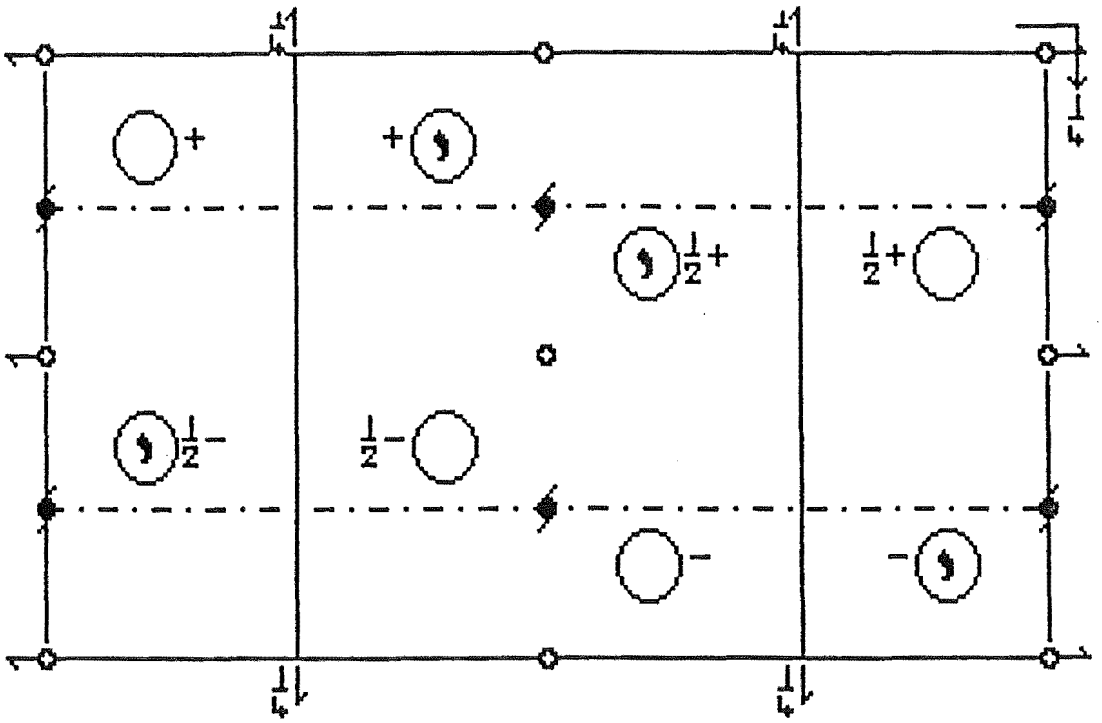


Figure 7.6 Diagram of space group Pnma.

equivalent position circle is placed on various positions intersected by some of the operations contained in Pnma. It is then shown how the operations containing no translational component relate the position back onto itself, while the operations containing a translation cause the position to be shifted. This concept has also been demonstrated using space group Cm which contains, apart from lattice centring, only two operations (see figure 7.3).

The display of molecules has been introduced by use of the crystal structure of either Benzamide [55] or Methionine [56]. Benzamide has the advantages of being in a space group familiar to the students ($P2_1/c$) and being a suitable example to illustrate the view modifying features (see figure 7.7). On the other hand the Methionine example gives an easily understood diagram immediately it is generated (see figure 7.8). In either case the strategy is to first generate the equivalent position diagram for the appropriate space group, then load the molecule and follow exactly the same steps in the generation of the diagram. In this way the correspondence between the equivalent position diagram and the unit cell of the actual structure may be clearly shown.

The effect of placing molecules on special positions may be illustrated using the structure of copper bis-acetylacetonate (file CUACAC), the copper atom of which lies on a centre of inversion in space group $P2_1/n$ (see figure 7.9). The notion that the asymmetric unit may consist of a fraction of a molecule may be illustrated in

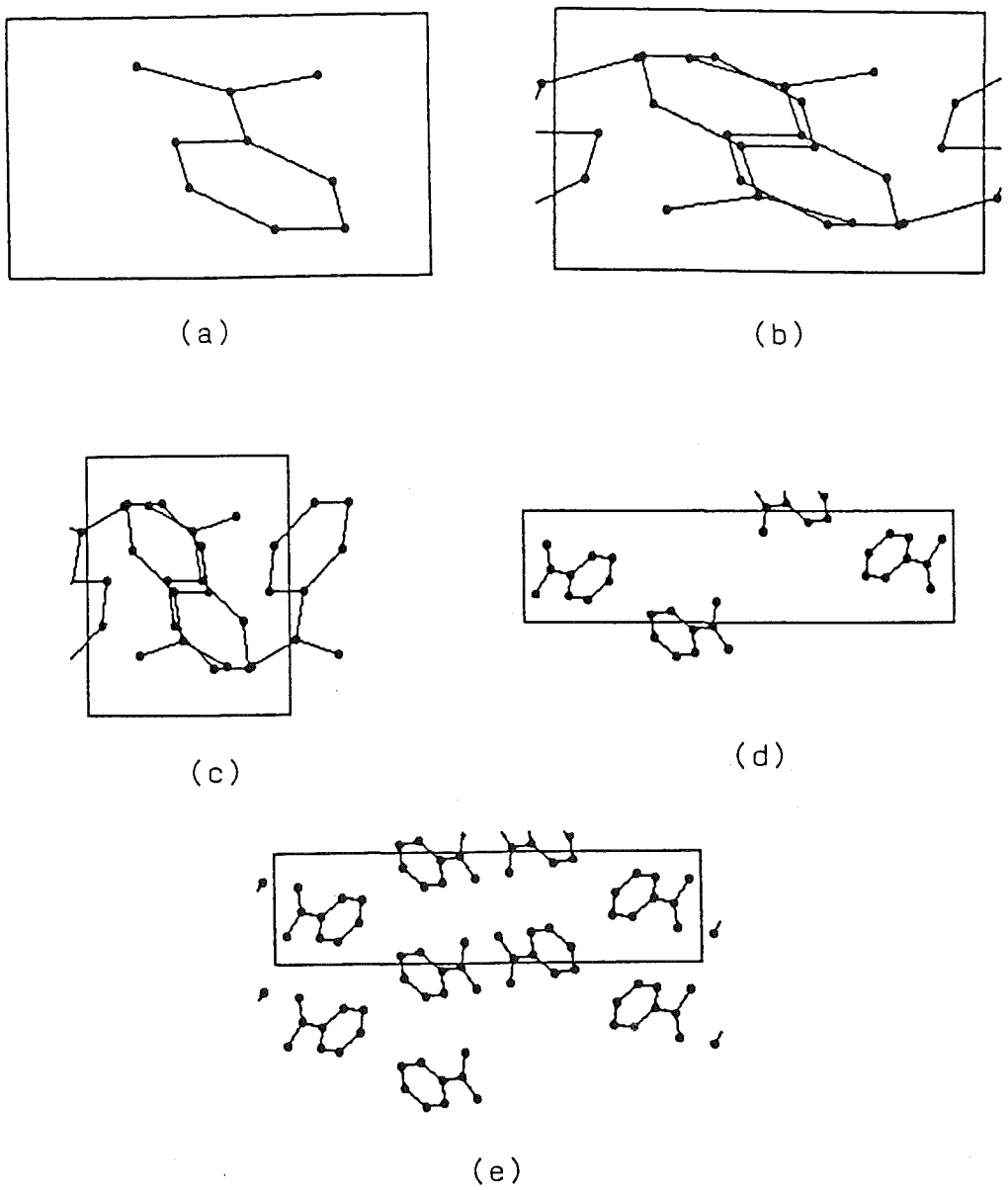
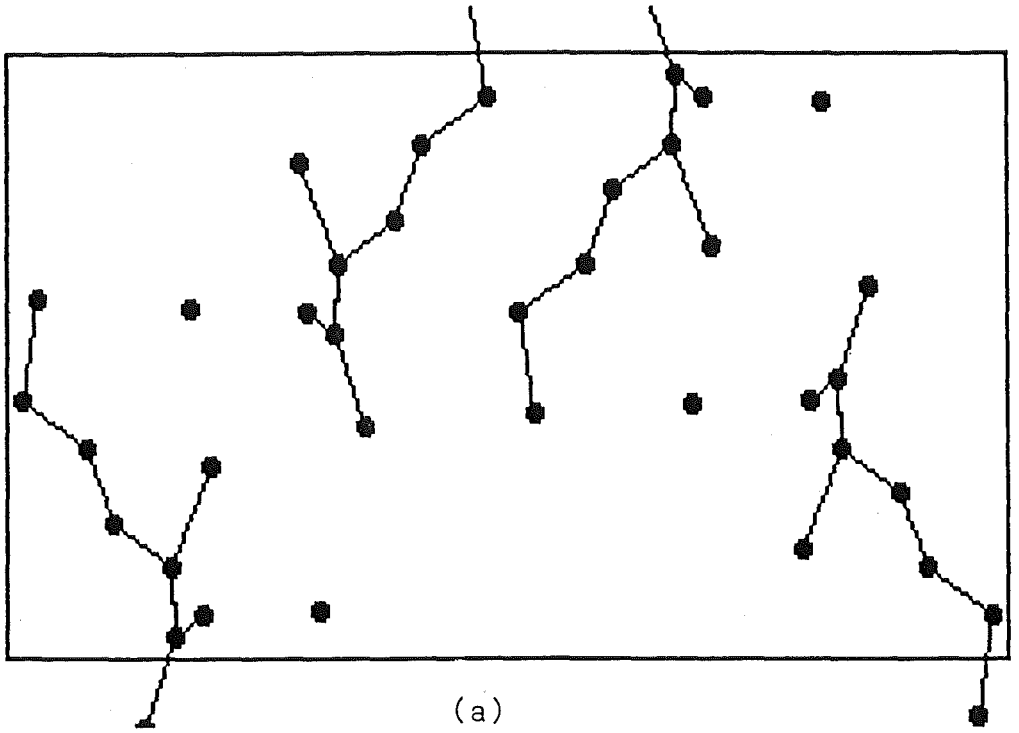
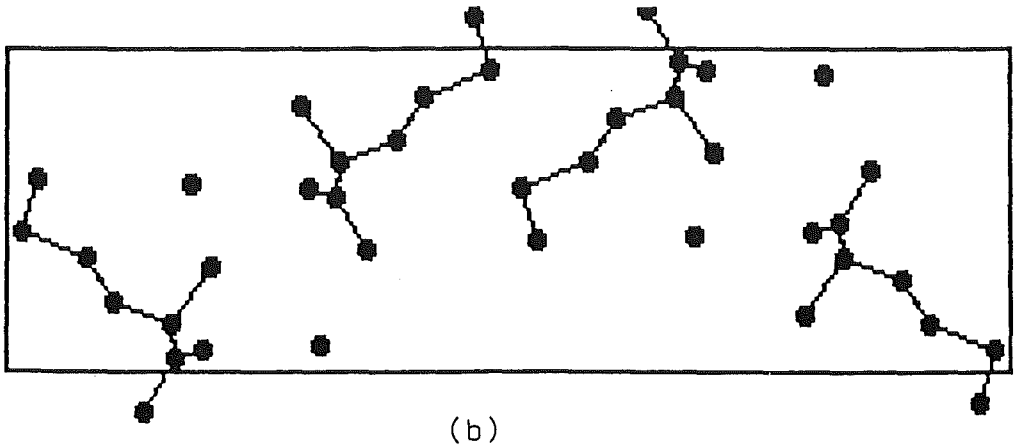


Figure 7.7 Steps during the building of a molecular display for the molecule Benzamide. a: The initial display. b: After the addition of symmetry operations. c: The unit cell dimensions are then added. d: The XYZ command is used to give a different view of the unit cell. e: The REP command is used to generate neighbouring unit cells.



(a)



(b)

Figure 7.8 Unit cell diagram for molecule file Methionine. a: As initially generated. b: After the addition of unit cell dimensions.

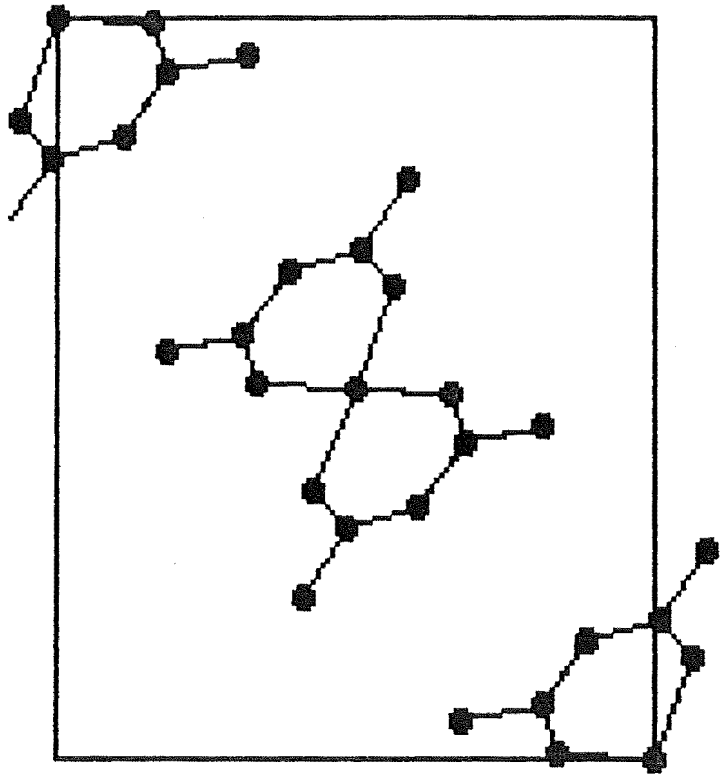


Figure 7.9 Unit cell diagram for molecule file Cuacac.
This diagram is a view along the b axis.

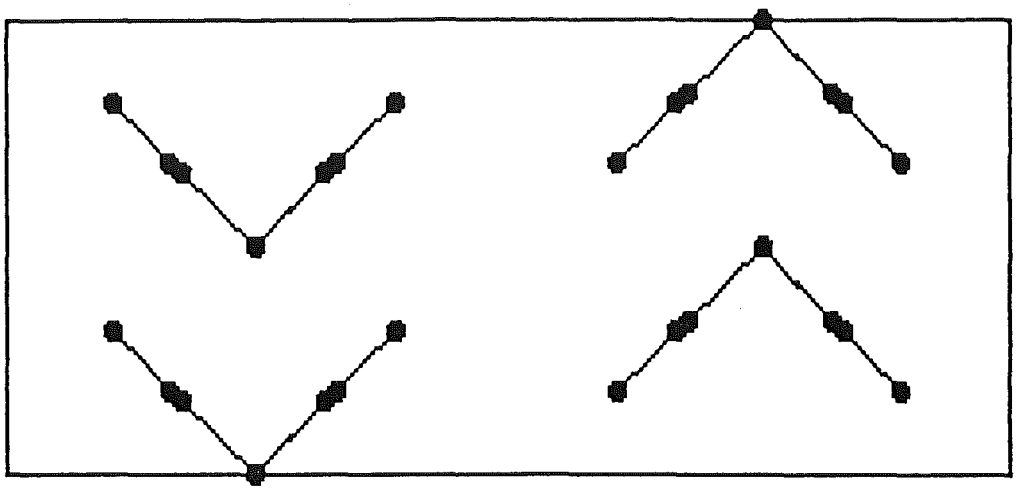
this case. A second example of this type used is the trans-trans form of diacetamide (file DAATT) [57], which contains an internal two-fold rotation axis, in space group Pbaa (see figure 7.10). For comparison purposes the cis-trans form is also displayed. This form crystallized in space group $P2_1/n$ and has no internal symmetry [58] (see figure 7.11).

7.6 USE BY INDIVIDUAL STUDENTS

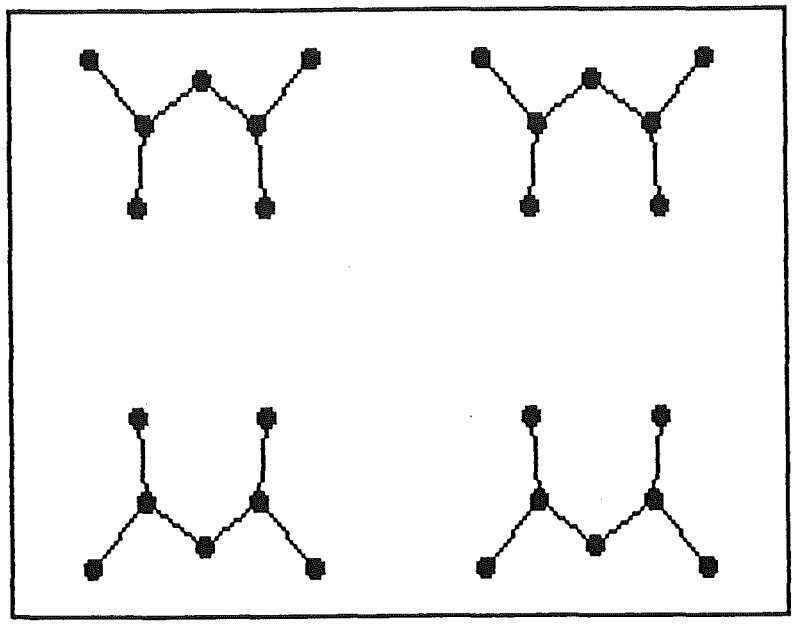
SGROUP may be used by students on an individual basis as a follow up to its use in a lecture demonstration. Students may either perform a series of prepared exercises, or explore the subject area more freely by experimenting with the capabilities of the program. Use of the program in lectures therefore has the added benefit of giving students some direction to the later individual use.

As a follow-up to the lecture demonstrations described in section 7.4, students are given a set of exercises to be performed, using program SGROUP, on an individual basis. As well as this set of exercises the students are provided with printed instructions for using SGROUP, and a set of space group diagrams. The instructions contain information on the way SGROUP is executed and how to enter operations, but they do not contain step by step information on the generation of space group diagrams.

Typically each exercise involves the generation of a standard space group diagram followed by the generation of



(a)



(b)

Figure 7.10 Unit cell diagrams for molecule file Daatt. a: A view along the c axis. b: View along the b axis.

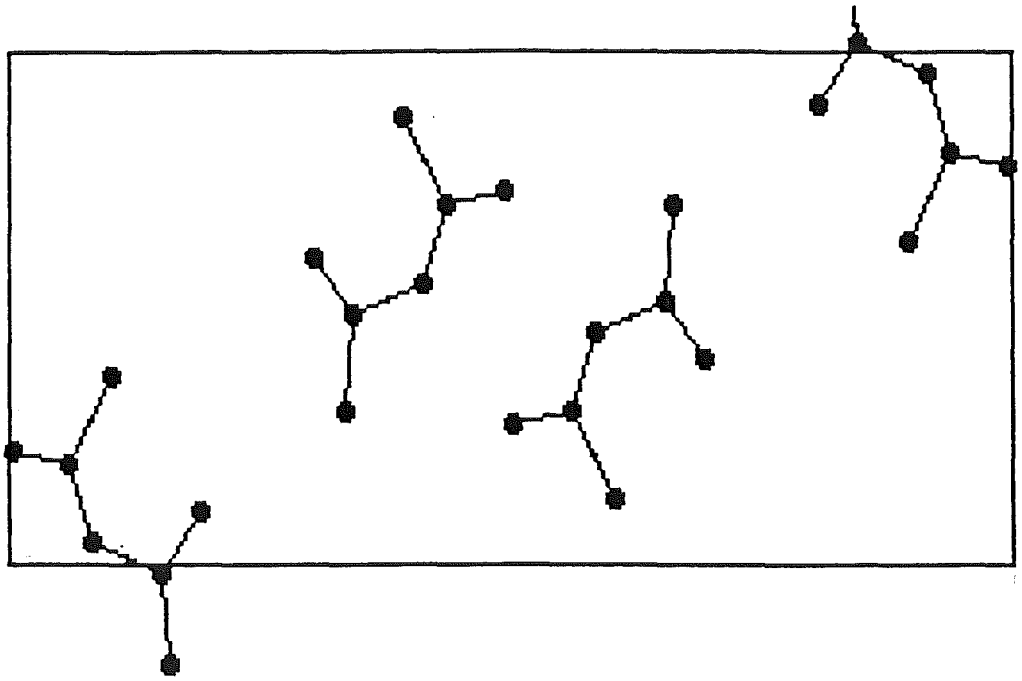


Figure 7.11 Unit cell diagram for molecule file Daact. The view is along the a axis.

one or more related diagrams. The final diagram would normally be a unit cell diagram of a molecule which crystallizes in the initial space group. Any intermediate diagrams would be used to introduce any special features present in the final diagram, such as a non standard orientation or an object on a special position.

Once a diagram has been generated, the student may then be asked questions relating to special characteristics of the diagram. The questions cover the packing of molecules within the unit cell, the orientation of the unit cell, the multiplicity of positions, asymmetric units and the most suitable way to view the unit cell. The purpose of the questions is not primarily to test the knowledge of the student, but rather to ensure that the student is aware of the features which should be noted when examining either a space group diagram or a diagram of a unit cell.

At the end of the exercises information is given relating to more structures available in the available package. The students are invited to build up the diagrams for the given structures, given the space group and the cell parameters for the structure. With the experience gained from the earlier questions a student should be able to note all the relevant features present in each diagram without prompting.

Experimentation with the capabilities of the program is an example of auto-elaborative CAL as described by Boyd [9]. This type of approach using SGROUP is most suited to teaching the effect of any operation and what parameters are

required to fully specify the operation. Although more structured, the setting of problems to be solved using SGROUP also falls within the auto-elaborative category as it is a way of directing the students towards a particular area while still letting them discover the principles themselves.

The more highly structured type of tutorial lesson, involving question and answer techniques, is not suited to an auto-elaborative type of approach as it does not allow students the flexibility to explore the subject in their own way. The question and answer tutorial or drill and practice lesson is more suited for use as a tool to test the level of comprehension gained by use of a program such as SGROUP. Such lessons have been written for use with SGROUP and are examined in chapter 8.

CHAPTER VIII

TUTORIALS FOR SGROUP

Once program SGROUP had been written the next step was to construct a set of related drill and practice lessons to be used by students for revision purposes. By use of such lessons students are able to determine the extent of their understanding of the information introduced in SGROUP. Such revision lessons may also be used by teachers to determine the degree to which a subject area is understood by the students who use the programs. With the addition of the drill and practice lessons to the SGROUP program, computer coverage of space groups extends from their introduction in lectures through to final revision.

8.1 PROGRAM DESCRIPTION

All the follow-up lessons were written in the MASTER tutorial language. The lessons constitute part of the two disk package of tutorial routines described in section 5 of appendix A, although the SGROUP related lessons could be accommodated on a single disk system if all disk space saving techniques were to be employed.

A total of four MASTER programs have been written for use as lessons to be used by students subsequent to use of program SGROUP. Three of these four are grouped so that they may be presented as a single lesson to a student using

the tutorial package. The three lessons have the same presentation format and collectively cover most of the ideas introduced in program SGROUP. The remaining lesson deals with some of the ways in which space group information may be used in crystallography.

8.2 THE "SPACE GROUP QUESTIONS" LESSON

The three MASTER programs which are collectively presented to students as "Space Group Questions", are all drill and practice lessons based on space group diagrams generated by SGROUP. The diagrams used are equivalent position diagrams, with labels and symmetry element symbols for each position. The presentation of the text part of the lesson is confined to the lower section of the display, replacing the prompt line and input line display area used in SGROUP (see figure 8.1).

The first of the lessons is saved on disk under the title SGSTAF, and contains questions relating to space group $P2_1/c$. The lesson contains twelve questions, each with a set of answers or definitions, the response displayed being dependent on the student reply to the original question. There is also provision for the presentation of hints following an incorrect answer so that the question may be attempted a second time. The correct answer is always displayed finally even if the student reply is judged to be correct. This is because the student reply might be a guess or it may contain elements of the correct answer in an

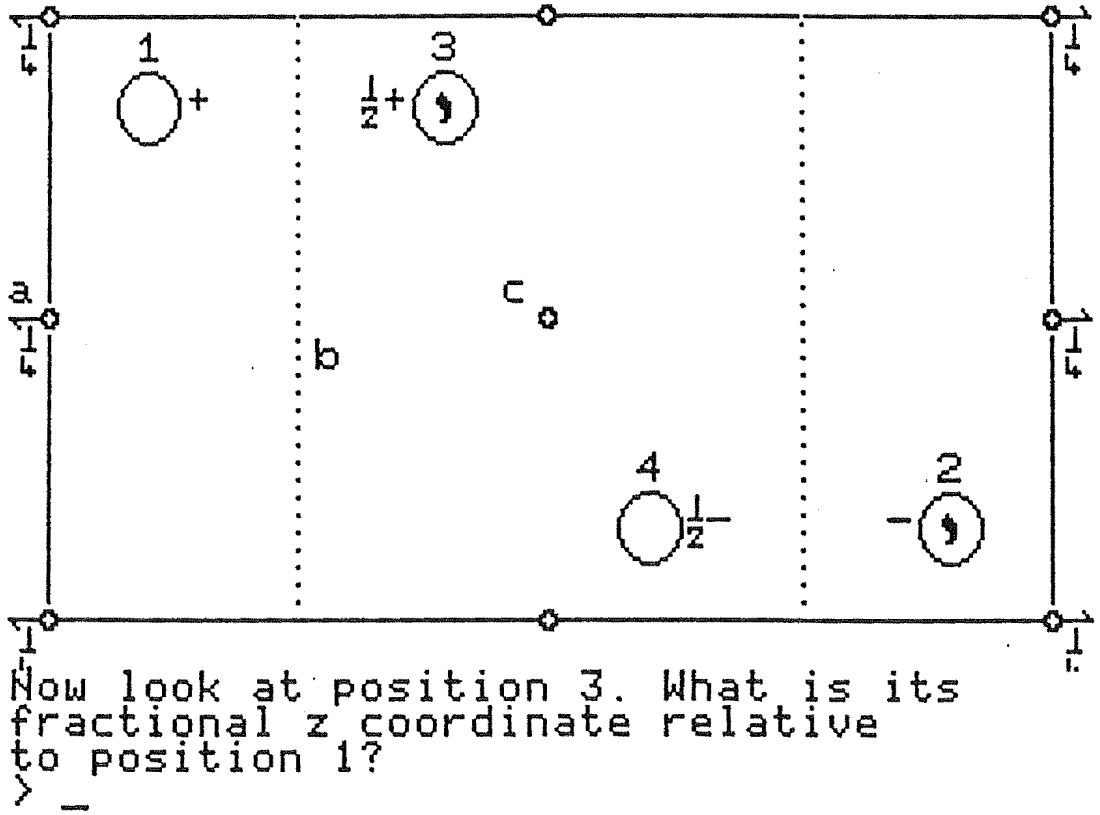


Figure 8.1 Sample display from MASTER lesson SGSTAF. This lesson constitutes the first part of the "Space Group Questions" lesson.

incorrect response which the response matching routines are unable to handle.

The SGSTAF lesson deals initially with the symbolism used to denote the various parameters used in a space group diagram. The first group of questions deals with the the handedness of positions, and with the representation of the coordinate along the c axis of the space group, which is the out of screen coordinate for the view of the space group used [53].

The second, and largest, group of questions in the lesson deal with the relationships between given pairs of equivalent positions. The students are asked to identify symmetry operations either by the way two positions are related to each other or by the symbol used to denote the operation. Further information is then required regarding the position or orientation of the symbol. The final questions in the lesson deal with space group multiplicity and special positions.

Since the space group used in SGSTAF is one that is studied in detail by students during the course prior to the running of the lessons, the students are expected to experience little difficulty with the questions. The questions are such that only a single aspect of the subject is covered at any stage. Multiple concepts and applications are avoided in order to provide the student with a gentle introduction to the way the system works while giving revision material on some basic material which will be of use in later lessons.

The second of the lessons making up "Space Group Questions" is contained in two disk files which are titled PMNA and PMNA2. The reason for the splitting of the program was the lack of sufficient computer memory to store the entire lesson at one time. This lesson contains thirteen questions based on a diagram of space group Pmna which is displayed and labelled in the same manner as the space group used in the previous lesson (see figure 8.2).

Space group Pmna contains examples of all the symmetry operations present in monoclinic and orthorhombic space groups. The relative complexity of this space group, compared to the space group used in the first lesson, means a greater number of questions are possible, together with a greater number of possible answers. As a result of the increased number of possible answers obtainable from the screen, and the greater complexity of the questions, more possible student responses are matched, with appropriate program replies.

The initial questions in the PMNA lesson deal with the way a symmetry operation acts on a general position. The description of a symmetry operation from its symbol is also examined during this first phase. The combination of two operations to give a third is the next subject treated, followed by the identification of special positions in the space group. The final section of the lessons covers possible multiplicities of positions and the effect this would have on the identification of internal symmetry in a molecule crystallizing in space group Pmna.

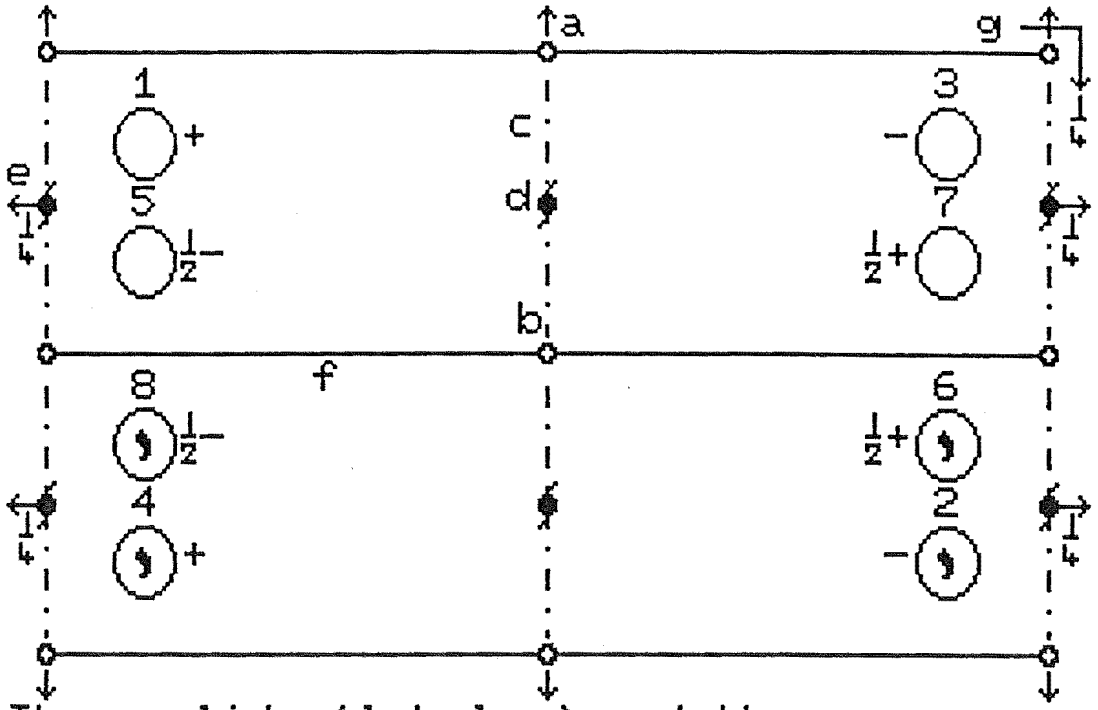


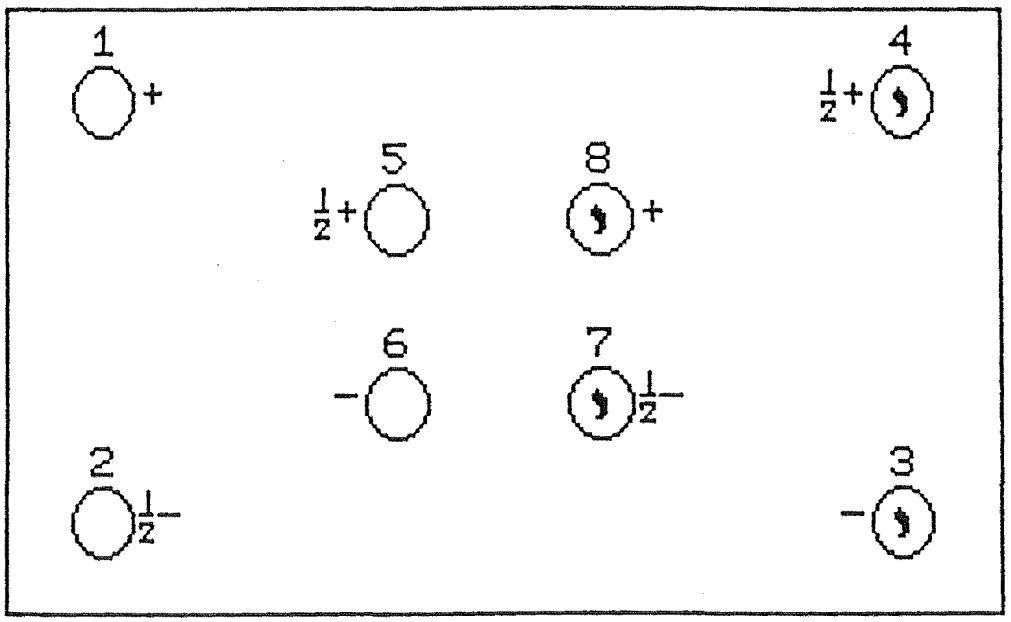
Figure 8.2 Sample display from lesson PMNA, the second section of "Space Group Questions".

The third lesson, named PBCN, is based on a diagram of space group Pbcn in which only the equivalent position circles are present at the beginning of the lesson (see figure 8.3). Individual symmetry operation symbols are added during the course of the lesson once the questions dealing with a particular operation have been presented and answered. The purpose of removing the symmetry operations symbols was to better determine the level of understanding of the equivalent position diagram.

The lesson PBCN consists of nine questions all of which are primarily concerned with the symmetry relationship between pairs of equivalent positions. The questions require the identification and characterization of symmetry operations either directly from the equivalent position circles or from other operations inferred from the circles. Questions are also asked relating to the space group symbol and how it may be used to determine which operations are present and their orientation. As with other lessons, there is positive feedback following initial incorrect answers.

8.3 THE "UNIT CELL QUESTIONS" TUTORIAL LESSON

The final drill and practice lesson of the programs, designed to complement program SGROUP, is named CCORD, although it is presented to students as the "Unit Cell Questions" lesson. This lesson is designed to be used after the use of the "Space Group Questions" lesson, as many of the questions require some familiarity with concepts covered



To which axis (a,b,c) is this rotation axis parallel?

> _

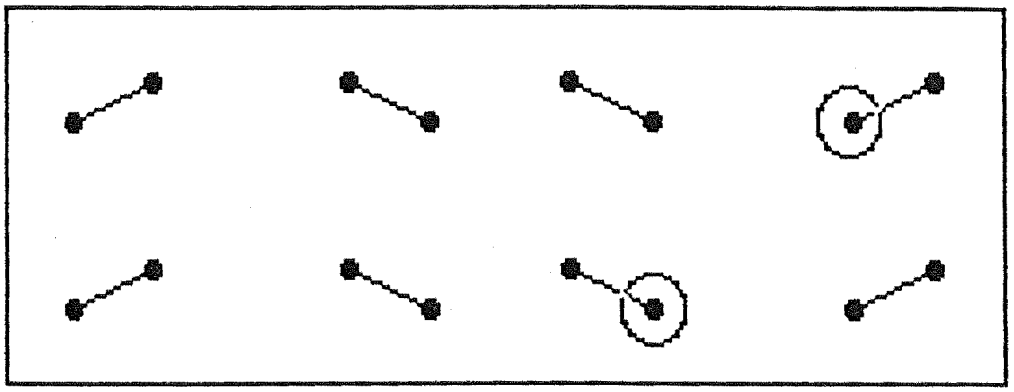
Figure 8.3 Display taken from lesson PBCN, the third and final part of "Space Group Questions".

in the earlier lesson.

The questions in CCORD involve the unit cell of a molecule which crystallizes in space group Pbca. The molecule used is not a real species but designed to best match the type of questions to be asked. The unit cell was constructed with space group Pbca symmetry because Pbca includes the same symmetry elements as those found in space group $P2_1/c$. The initial questions in the lesson are based on $P2_1/c$ symmetry to give the student a familiar framework in which to work, as was done in the SGSTAF lesson. Although all the questions are able to be answered using the information contained in the text presented in the lesson, a diagram of the unit cell is given at the top of the screen to give the student a physical picture of the subject of the question (see figure 8.4). The diagram is built up during the course of the lesson to show the effect of the symmetry operations being considered. During the lesson atoms may also be marked by circles to indicate that those particular atoms are the subject of the current question.

The lesson consists of eight questions, together with hints and supplementary questions which are given when the response is incorrect or contains only part of the correct answer. The topics dealt with are the relationship between fractional coordinates and coordinates given in angstroms, the effect of applying a symmetry operation to an atom, the determination of symmetry when given coordinates, and deriving symmetry from the space group symbol.

Most questions in the CCORD lesson involve some



By what operation are the atoms at
0.8,0.65,0.26 and 0.3,0.85,0.74 (frac)
related?> SCREW
Yes they are related by a screw. To what
axis is this screw parallel?> A
Correct : they are related by a screw
axis parallel to the a axis.

Press <RETURN> to continue > _

Figure 8.4 Sample display from lesson CCORD. This diagram includes a complete dialogue for a specific question.

numerical calculation to be carried out by the student. To give an impression of realism, and to avoid student guessing of trivial answers, the calculations were constructed so as not to be too simple in nature. However it was also important that calculations not be so complex that students could not complete them mentally, even though the introduction to the program notes that a calculator may be of benefit during the lesson. The result is that no number supplied or answer expected was to contain more then three significant figures or two decimal places. Keeping to these restrictions also simplifies matching of answers. However some rounding was allowed for in answer matching as was the possibility that students may enter the calculation to be performed rather than the numerical result of that calculation.

8.4 ADMINISTRATION OF LESSONS

All arrangements for use of the SGROUP related drill and practice lessons were the same as those for SGROUP itself so that students could go from using one to the other with the minimum of inconvenience. Instructions for use of the MASTER system were provided either by a lesson named KEYBD or by a page of written documentation. Unless the student possessed no knowledge of keyboard skills, the only information required by students was how to start the system and the format required of responses to questions. If an introduction to the use of the keyboard was required this

was provided by supplying the student with the KEYBD lesson or the Apple //e introduction lesson.

At the completion of a MASTER lesson all student responses are saved on a disk file for subsequent analysis. Examination of the response files is important in determining if students are experiencing difficulty with any question or concept. A large number of incorrect replies to a question may indicate that the students do not understand the concept being considered, or that the question is badly phrased so that students misunderstand it. It is important to follow up questions which draw unexpected responses so that the source of the problem may be identified and remedied. Examination of response files also serves to identify those responses which are correct answers to a question but have not been flagged as such by the matching routines in the program.

During the development of the programs it was useful to observe subjects running the MASTER lessons and the problems they encountered. Program bugs or inadequacies, not foreseen during the writing of the program, are often uncovered by observation of the program in use. Seeking the opinions of subjects upon their completion of a lesson also gives an insight into the way the program is perceived by the student. The examination of responses and observation of program users constitutes a significant proportion of the development time for any tutorial or drill and practice program, which is why such programs generally take more time to develop than lecture demonstration style programs.

CHAPTER IX

THE PATTERSON PROGRAM

The general expression for the electron density in a crystal structure is the Fourier summation:

$$\rho(x,y,z)=\frac{1}{V}\sum_{hkl}\sum\sum F(hkl)\ \exp\{-2\pi i(hx+ky+lz)\}$$

where F is a complex quantity with a magnitude and a phase, both of which must be known if the electron density in a crystal is to be evaluated. Another Fourier summation, the Patterson function, is defined as:

$$P(u,v,w)=\frac{1}{V}\sum_{hkl}\sum\sum F^2(hkl)\ \cos\ 2\pi(hu+kv+lw)$$

where F^2 values are proportional to measured intensities [59]. The Patterson function is specially important as a means of solving crystal structures because it may be evaluated directly from experimental diffraction intensity data. It may be formally considered to be the convolution of the electron density with itself and can be physically represented as a vector map, where each peak in the map corresponds to the vector between two peaks in real space, a form which should be readily comprehended by the student.

Program PATTERSON was written for use as a tool in the teaching of the Patterson function by using the computer's graphics display to represent the correspondence between the arrangement of atoms in real space and the peaks in vector space. A number of features were written into the program to highlight particular properties of the Patterson function.

9.1 PROGRAM DESCRIPTION

The screen display is divided into three parts. The largest area of the screen is taken up by two rectangles which may be taken to represent projections of the unit cell of a crystal. That on the left represents real space, with small filled circles indicating atom positions. The second rectangle, on the right of the screen, represents "Patterson" or vector space. The peaks are indicated by circles with a larger radius to indicate the greater size of the vector peaks than their counterparts in real space, electron density peaks.

The third part of the display is a line of text at the bottom of the screen which may contain either a prompt line or warning and error messages (see figure 9.1). The prompt line contains the letters corresponding to the valid program commands (see table 9.1). A command is activated by pressing the relevant letter.

Initially the contents of a unit cell are entered, either by loading a file from disk or by entering the coordinates from the keyboard. Provision is made to add symmetry to the unit cell once an asymmetric unit has been entered. The operations are the same as those available in program SGROUP although the display is more limited. No provision is made for bonds between atoms, and symmetry elements are displayed simply as lines or circles to show their positions (see figure 9.2).

Once a unit cell has been entered a vector map may be

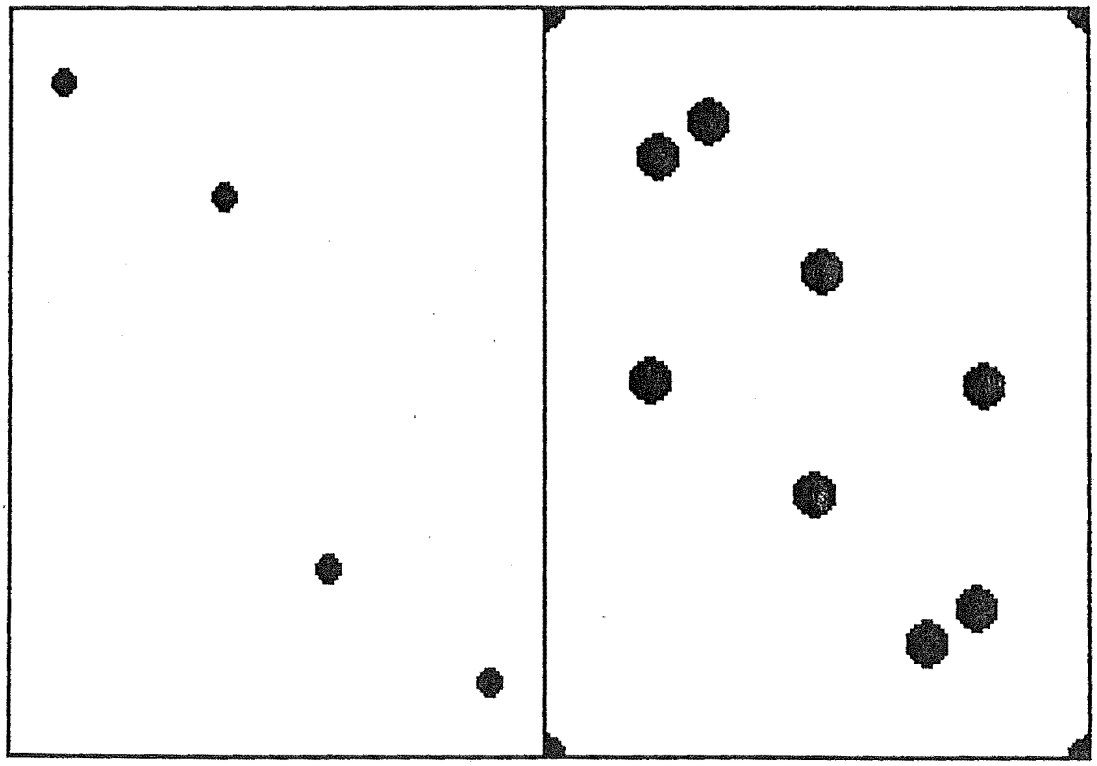


Figure 9.1 Sample display for program PATTERSON. The section on the left represents the unit cell while the right hand side represents the vector map. The line of text at the bottom of the display is the program prompt line.

Commands for Patterson Plot

code	action
M	Menu of operations
N	New cell (from keyboard)
Y	Read cell from disk
Z	Write cell to disk
D	Draw cell on screen
C	Coordinates of atoms in cell
L	Label atoms
!	Label peaks
T	Transpose origin
K	Symmetry operations
V	Generate vector map
B	Display a single vector
H	Indicate peaks on Harker sections
W	Show peak weightings on vector map
O	Place cell on origin of vector plot
S	Superposition of vector peaks
P	Positions of vector map peaks
X	Examine peaks (single/double weight)
R	Rotate x,y,z coordinates
I	Interchange x & y coordinates
G	Graphics dump to printer
E	Exit program

Table 9.1 Commands for program PATTERSON.

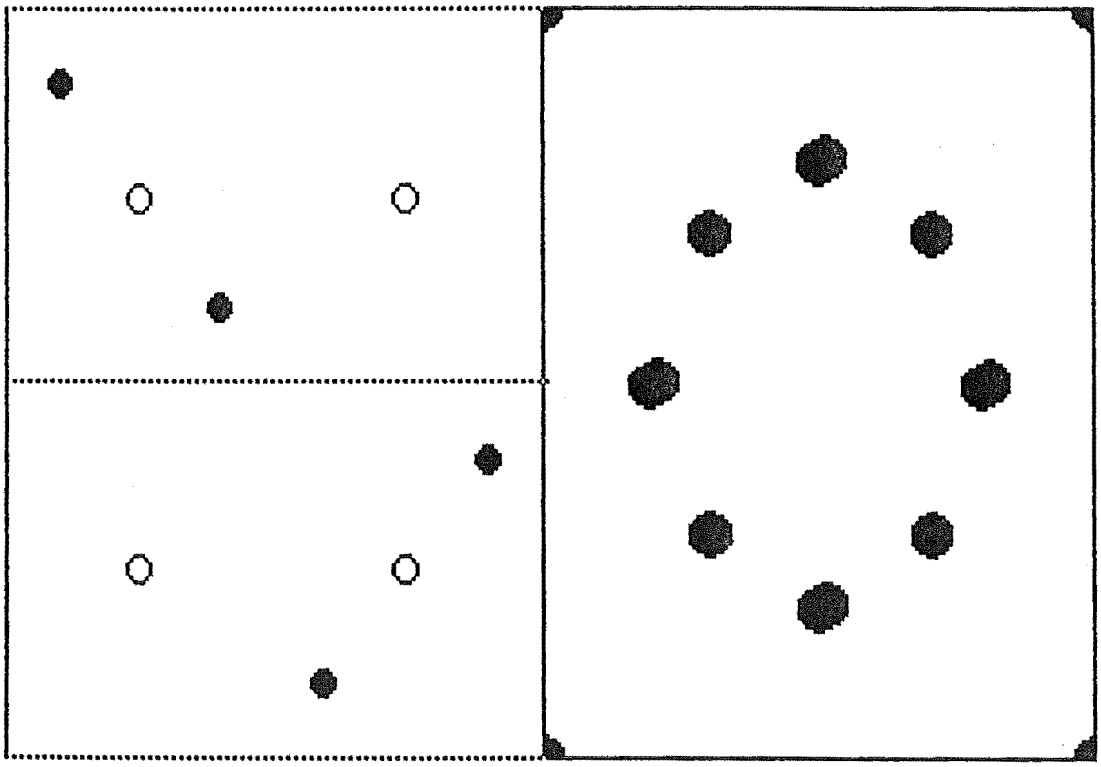


Figure 9.2 Display including symmetry symbols in the unit cell diagram.

generated by calculating the interatomic vectors for each pair of atoms in the unit cell. Each vector is displayed in turn as a flashing line between the corresponding atoms in the unit cell display. Simultaneously another flashing line is displayed from the origin of the vector map and a circle is generated at its end to indicate a "vector peak". As a teaching aid both the atoms in real space and the peaks in Patterson space may be labelled, and a flashing image of the unit cell may be overlayed onto the Patterson space display with a chosen atom placed on the origin (see figure 9.3). Further aids include the marking of Harker planes and lines, and the indication of peak weights (see figure 9.4).

9.2 PROGRAM REQUIREMENTS

Patterson was written in Apple-UCSD-Pascal 1.1, running under the Apple language system. To execute the program a data file, containing shape information for the atom and vector peaks, must be on line during program initialization. The program also requires a modified system character set if any symmetry element information is to be displayed. The special character set used in the symmetry operations package has had the appropriate changes made to it.

The program is designed to run using the minimum hardware configuration, with an optional printer for bit image graphics dumps. A second disk drive may be used to store files containing unit cell information if there is not space for them on the boot disk. A colour monitor is also

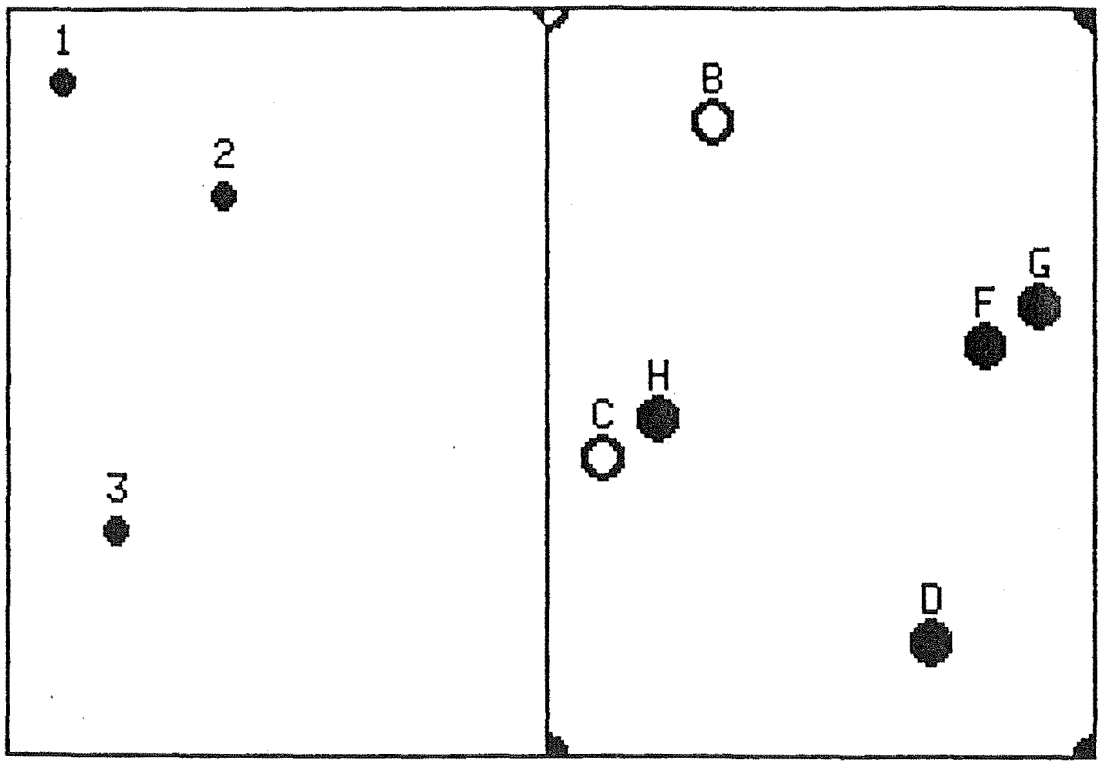


Figure 9.3 Display including labels for atoms in both the unit cell diagram and vector map. Also pictured is the effect of the cell overlay command which produces the cleared areas in vector peaks B and C as well as the origin peak.

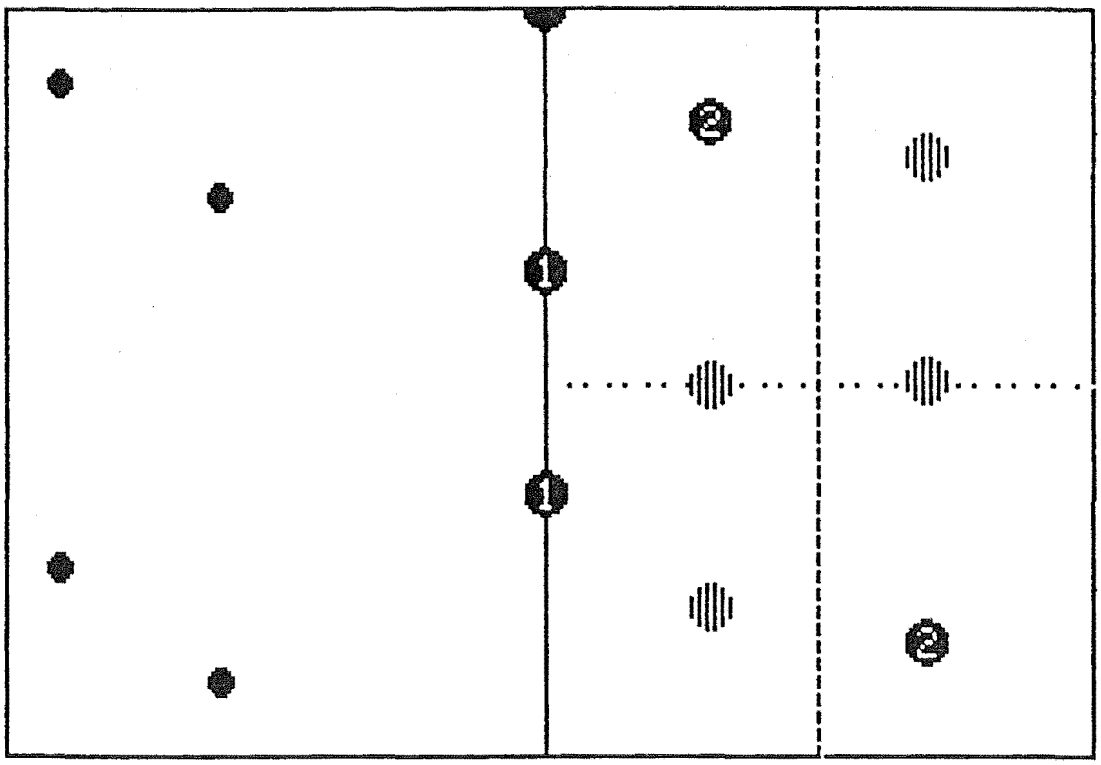


Figure 9.4 Display with Harker sections highlighted on the vector map. Peaks lying on Harker planes normal to the c axis are designated by a colour filled circle. The colour is represented as an area of vertical lines in diagrams or on monochrome monitors. Those peaks not on Harker sections retain the peak weighting indicators generated using the W command prior to the generation of the Harker display.

useful as a number of the features are enhanced by the use of colour in the display.

Documentation is provided for program PATTERSON as part of the symmetry operations package in which the program is included. Documentation consists of a disk file containing information on program execution and the usage of all commands.

9.3 TEACHING USE

Program PATTERSON is designed to demonstrate the concept of the Patterson function as a vector map, and how this interpretation of the function may be used as a tool in the solution of crystal structures. The split screen display enables the unit cell and the vector map to be visible together on screen, so that the relationship between them may be perceived by the viewer. This relationship is demonstrated most clearly at the time the vector map is generated as the flashing lines used to symbolize the vectors appear in both maps simultaneously. Because of the simultaneous generation of the vectors, the equivalence of the atom to atom vectors in real space and the origin to peak vector in Patterson space may be clearly shown (see figure 9.5).

The ability to alter the unit cell contents gives flexibility in teaching. A simple structure, say of two atoms, may be used to demonstrate the basic concept of the vector map, while a more complicated arrangement may be used

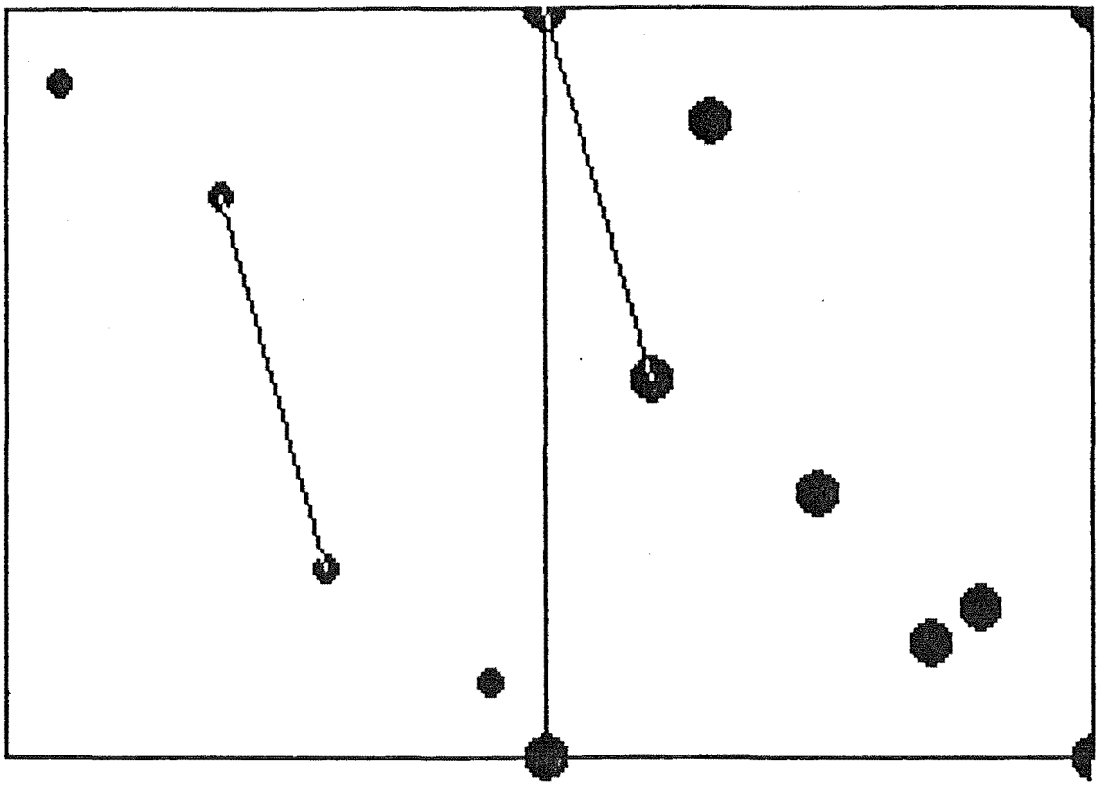


Figure 9.5 Diagram taken during the process of building a vector map. The lines, representing vectors, joining the circles flash rapidly during program execution.

to show the difficulties that arise in actual experimental situations where, in two dimensions, significant overlap of peaks occurs. The effect on the vector map of the presence of symmetry in the unit cell may be shown by the use of the symmetry generation routines and the provision to mark Harker sections.

Other manipulation of the unit cell for teaching purposes is also possible. The orientation of the cell may be changed so that views down any crystallographic axis are possible. The unit cell origin may be shifted from the corner of the display to the centre, to demonstrate that a vector map generated from within a particular unit cell is the same as that generated when the vectors cross cell boundaries (see figure 9.6).

It is important to show how the Patterson function may be used as a structure solving tool. By overlaying views of the unit cell diagram on the Patterson map diagram it is possible to show that the Patterson map contains images of the atom arrangement (structure) which may be extracted during a crystal structure analysis. The simulation of the superposition method may then be used to show how this is actually done (see figure 9.7). A number of different superpositions may be carried out to show the effect of different types of vector peaks, the peak weighting function being used to distinguish the various types of peaks.

Program PATTERSON was initially designed as a lecture demonstration tool, with a flexible mode of operation to allow the teacher to demonstrate clearly as many aspects of

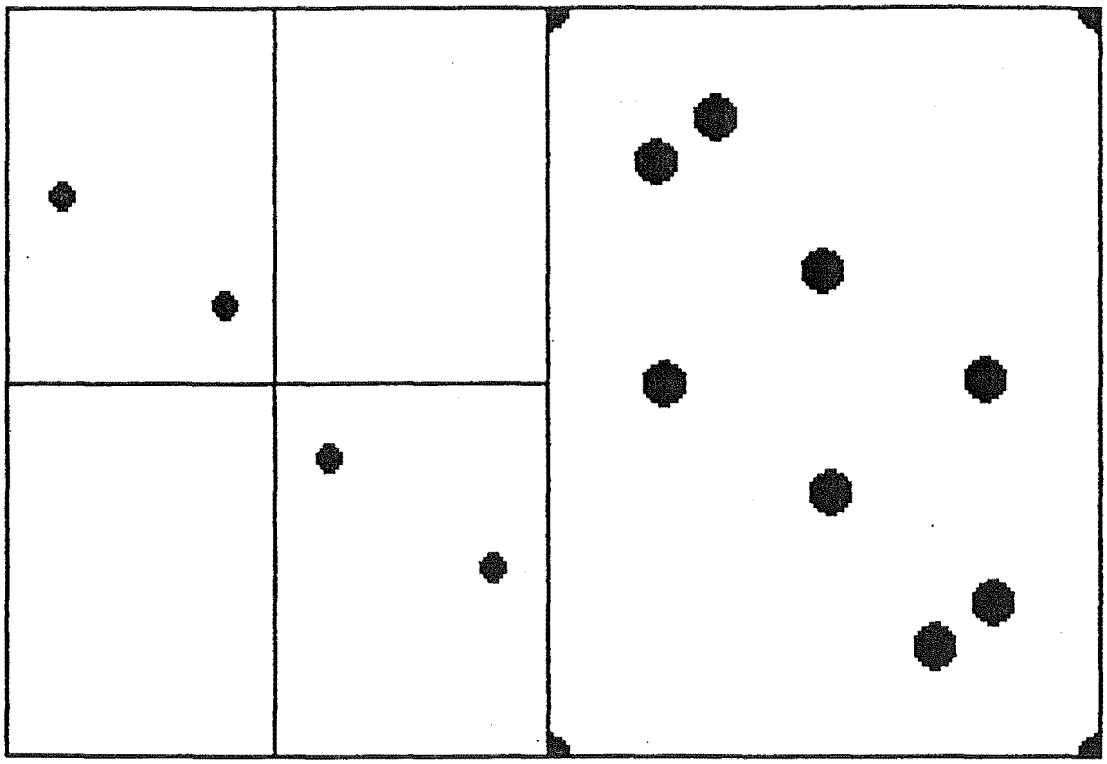
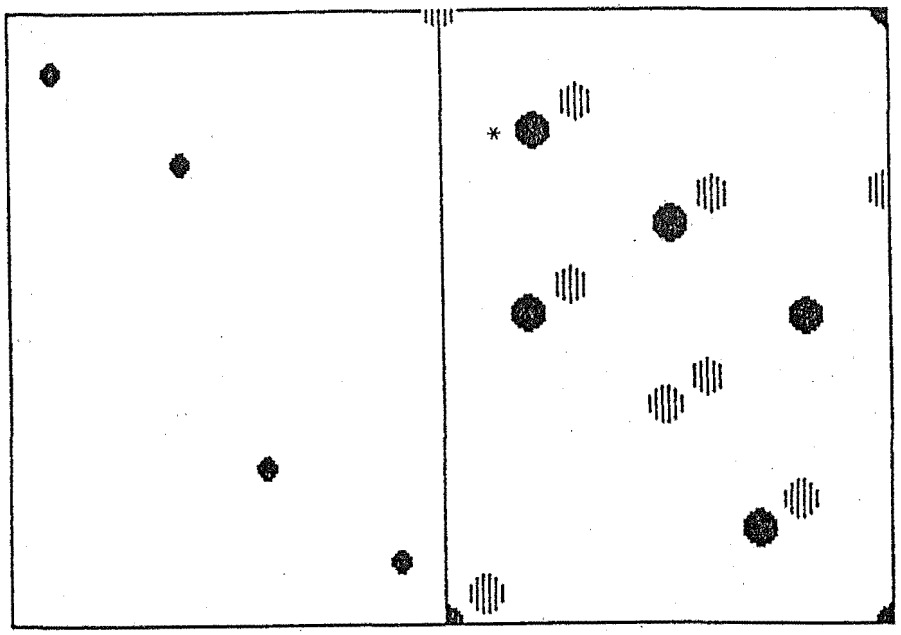
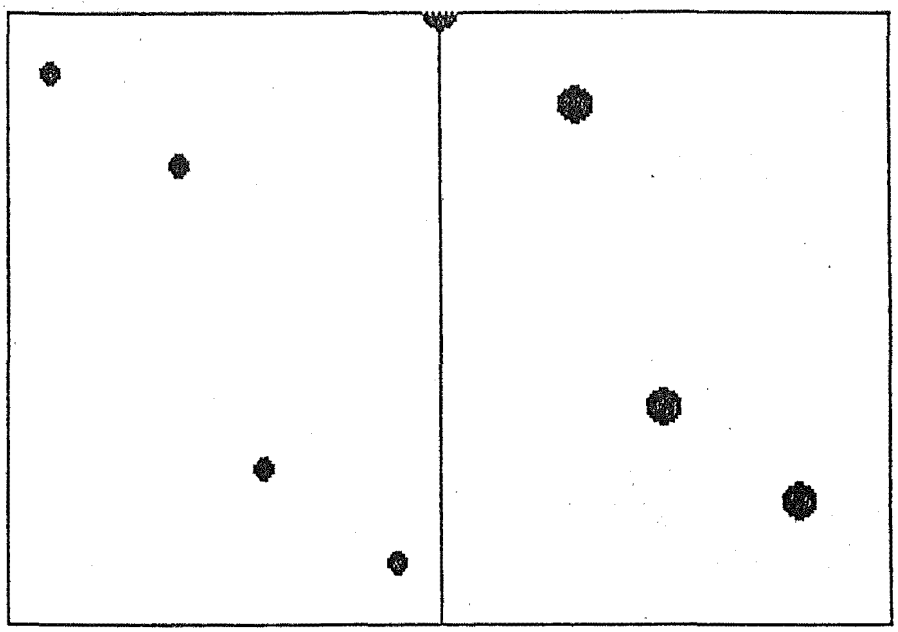


Figure 9.6 Program display with the unit cell transposed. This diagram is the same as figure 9.1 except that the unit cell origin is now in the centre of the unit cell display area.



(a)



(b)

Figure 9.7 The vector superposition simulation. a: Display during the course of superposition. A second image of the vector map has been added to the original map (figure 9.1) shifted by the length of the vector marked by an "*" symbol. The shifted vector map peaks are represented by the shaded circles. b: The resulting diagram after the completion of the superposition demonstration. The remaining vector peaks are those positions where superposition of peaks took place. Note that they represent an image of the unit cell.

the Patterson function as possible. However the flexibility of the program also makes it suitable for use by an individual or by small groups of students.

Unsupervised use of program PATTERSON by students may take place in the same manner as student use of program SGROUP. The teaching method used for SGROUP, as described in section 7.5, is an auto-elaborative approach which may also be used for PATTERSON by the provision of a similar set of instructions, examples and problems. Provision for auto-elaborative learning is made through the instructions on the use of the program and a set of varied atom arrangements which are available for use with PATTERSON.

9.4 AREAS OF USE

Program PATTERSON has been used in the University of Canterbury chemistry department during courses for the teaching of techniques of crystal structure solution. The BSc honours part II and part III courses contain sections covering the theory and practice of the determination of crystal structure. The program is used to introduce the Patterson vector map as a tool for determining the position of atoms in real space.

The BSc honours II class were introduced to the Patterson function during the course of practical laboratory sessions which covered basic aspects of crystallography. The class was divided into groups of 4 to 6 students with each group taken through the crystallography course

separately. Once a group of students had been supplied with supplementary material covering the theory of the Patterson function, program PATTERSON was used to provide a demonstration of the practical application. A vector map was generated to illustrate the physical nature of the Patterson map. Images of the structure were then highlighted in the vector map. A vector superposition was then performed to show how the structural information may be extracted. Demonstrations involving added symmetry in the unit cell and Harker sections were also included. All the unit cell information used in the demonstrations was taken from structures examined by the students in subsequent sections of the course.

Use of program PATTERSON with the BSc honours III class involved lecture demonstration and subsequent student use. The lecture demonstrations involved the concepts covered in the honours II demonstrations as well as covering peak weighting, peak sizes, the number of vector peaks and the presence of origin peaks. The presence of symmetry in the unit cell and its relation to Harker sections was covered in greater detail, as was the superposition method. Students subsequently using the program on an individual basis were supplied with instructions and sample unit cell information. No attempt was made to guide the direction of study by use of drills or set problems, but rather the students were invited to examine the properties of the Patterson function with particular emphasis on the effect of symmetry in the unit cell.

CHAPTER X

SYMMETRY OPERATIONS

When attempting to teach aspects of symmetry and how it relates to crystallography a necessary first step concerns the nature of the basic operations used to describe symmetry in molecules and crystals [60]. Although they may also be treated mathematically, the elementary symmetry operations have a physical meaning which may be demonstrated using computer generated graphics. This permits the use of animation and colour in the display, greater flexibility for the teacher in a class or lecture, and possible subsequent use by individual students.

10.1 THE SYMPACK PROGRAM

The demonstration of the elementary crystallographic symmetry operations was achieved through program SYMPACK. This program, as with other programs in the Symmetry Operations Package, is written in Pascal for the Apple computer. The program consists of six demonstrations covering reflection, inversion, glide, rotation and two views of the screw operation. All the demonstrations are simple, involving the application of a symmetry operation to a single object, so that the nature of the operation may be clearly shown.

The demonstrations of reflection and inversion consist

of "before and after" scenes in which an object is initially displayed alone on the screen and then, following a keystroke, together with the resultant image from the operation. In the case of reflection the object operated on is a question mark (see figure 10.1), while a distorted tetrahedron is used for the inversion since, for this operation, more three dimensional information is needed (see figure 10.2).

The remaining four demonstrations use animation to illustrate the nature of the movement of an object as it undergoes a translation or a rotation. The demonstration of the glide operation consists of a reflection followed by a translation parallel to an arbitrary axis across the screen (see figure 10.3). No attempt is made to demonstrate glides in other orientations or with more than one translational component since the purpose of the demonstration is not to give a detailed description of all possible glide planes, but simply to introduce the idea that a glide is composed of a reflection and a translation.

In the case of the rotation and screw axes a more complex demonstration is required since more than one order of rotation is possible. The demonstrations are made up of the animated rotation of a triangular shape, with the rotation being of order one, two, three, four or six (see figure 10.4). In the case of the screw axis demonstrations the rotation is followed by an animated representation of translation. The images produced by the rotation or screw operations may in turn be operated on so that the effect of

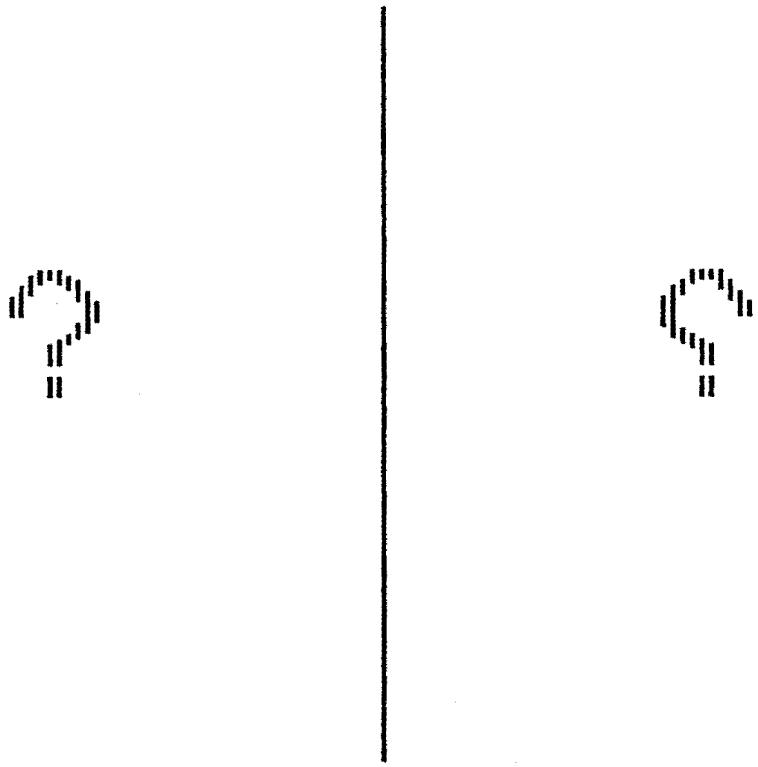


Figure 10.1 Reflection demonstration from program SYMPACK.

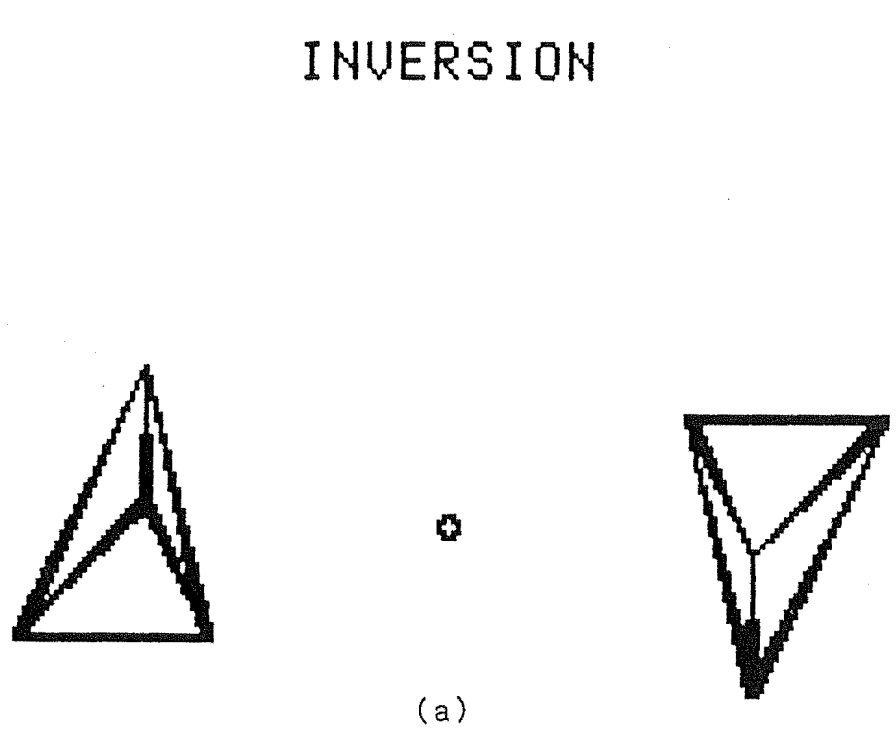


Figure 10.2 Inversion demonstration from program SYMPACK. a: The initial inversion demonstration. b: After lines connecting the tetrahedron vertices are added.

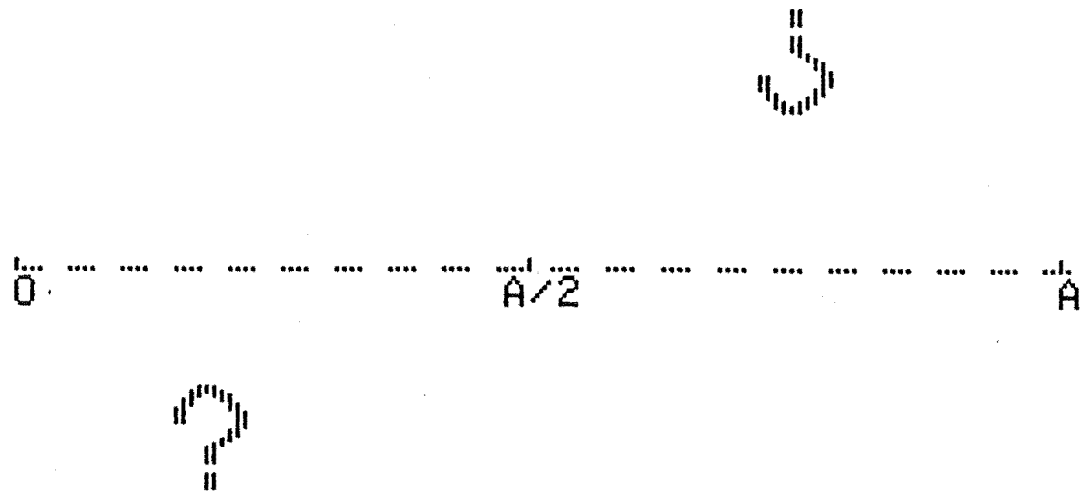


Figure 10.3 Demonstration of an a-glide from program SYMPACK. This diagram is the end result of an animated sequence.

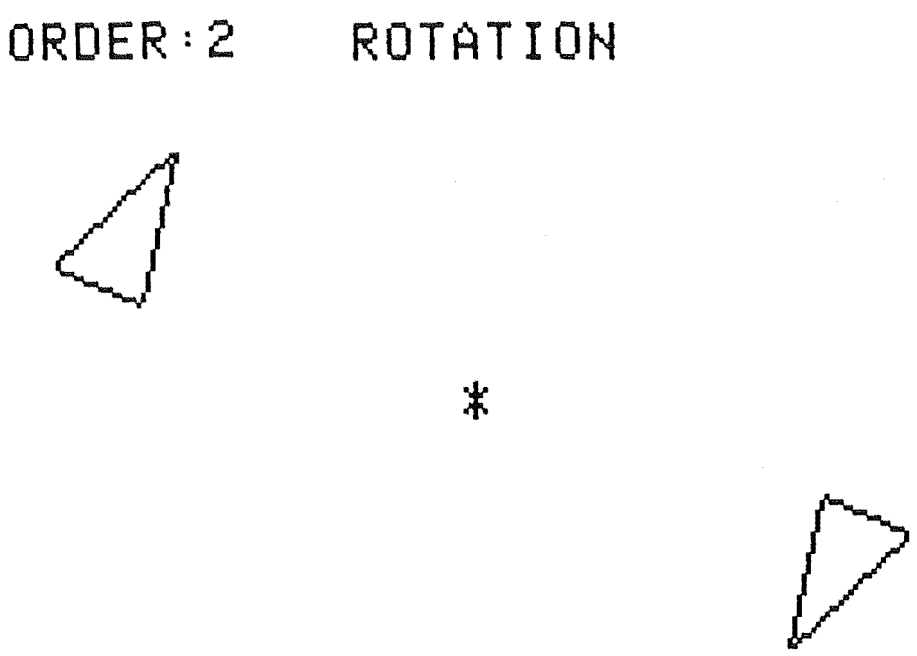


Figure 10.4 Final display after animated demonstration of two fold rotation about an axis normal to the viewing plane.

repeated application of an operation may be shown (see figure 10.5). Screw axis demonstrations are given in two orientations to ensure that the best view of each of the two components is available. A view parallel to the axis shows the rotational component, while the translational component is shown to greatest effect with a view perpendicular to the axis (see figure 10.6).

Program SYMPACK was intended specifically as a lecture demonstration aid which could present a variety of animated diagrams to demonstrate basic symmetry operations in a way more convenient to the teacher than other demonstration methods. The use of animation is a feature which can be matched by film or video demonstrations, but these media do not give the flexibility to change the sequence or number of demonstrations shown as is possible with a computer graphics based demonstration. As with other programs designed initially as lecture demonstrations, SYMPACK also proved to be suitable for use by individual students who used the program to reinforce the concepts presented by the computer during lectures. Unlike the individual use of programs SGROUP and PATTERSON where the emphasis was on auto-elaborative learning of new concepts prompted by a set of prepared problems, the use of program SYMPACK by individual students introduced or developed no new concepts but rather it gave the student the chance to become more familiar with the material which had been presented during lectures using the same program.

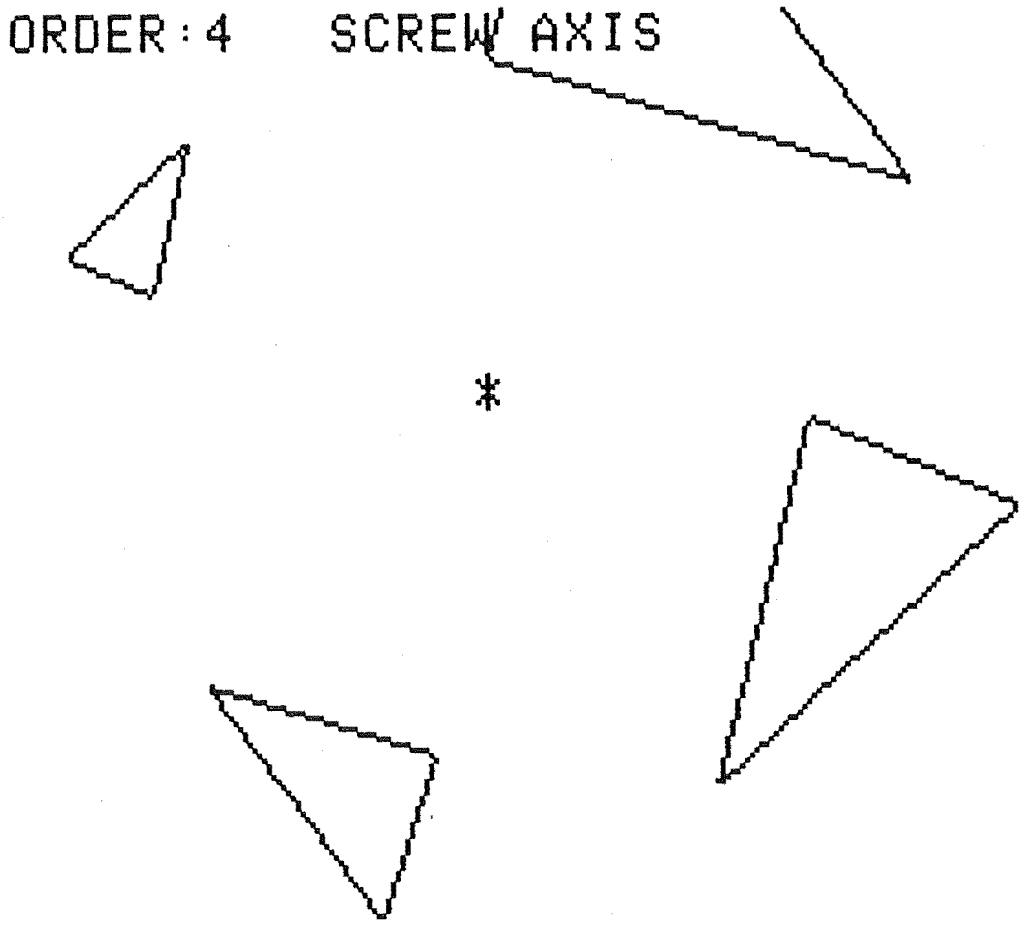


Figure 10.5 Demonstration of four fold screw axis viewed along the axis of the screw. This diagram is the end result of an animated sequence where the translational component of the screw is represented by an increase in the size of the triangular shapes being operated on.

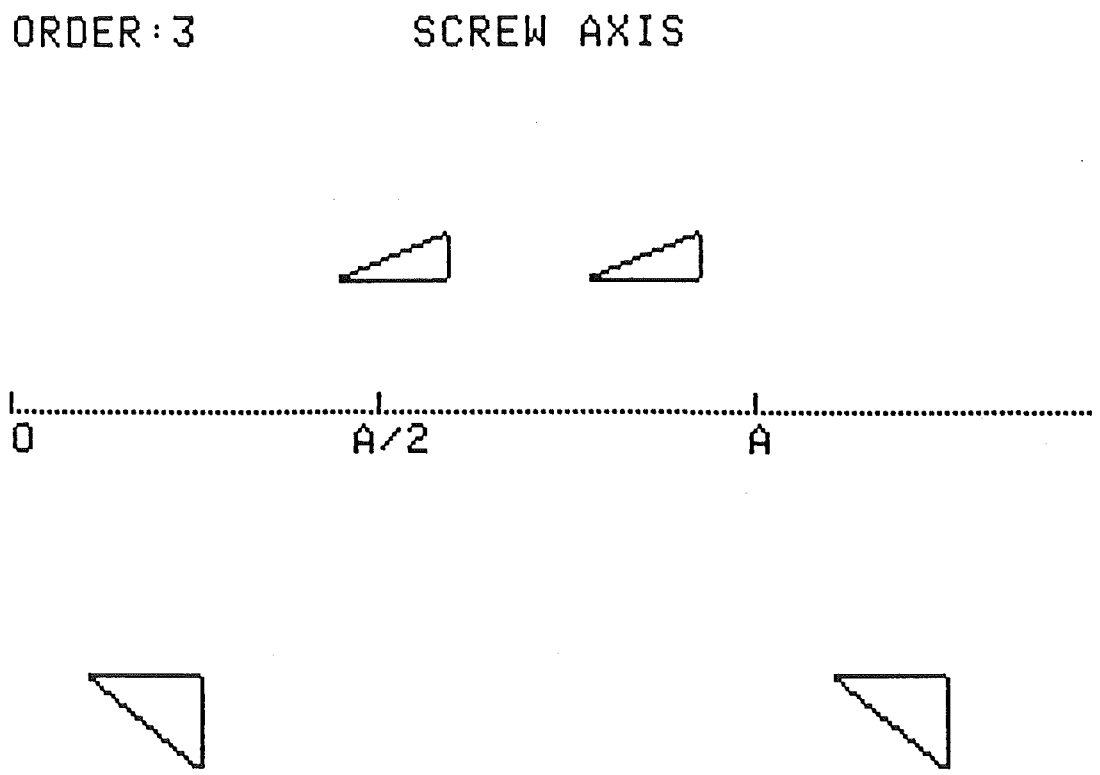


Figure 10.6 The end result of an animated demonstration of a three fold screw axis as viewed perpendicular to the axis of the screw.

10.2 THE TUTORIAL PROGRAM

As an alternative to the use of program SYMPACK as an introduction to basic symmetry operations, a tutorial program covering the same general subject area was written. This tutorial program, named TUTORIAL, used the same animated graphics routines as program SYMPACK but in this case the student user has little control over what is displayed. The demonstration of basic symmetry operations is given in a fixed order and is accompanied by explanatory lines of text which are displayed as the demonstration progresses (see figure 10.7).

Since the animated graphics routines could not be integrated into the MASTER system, for space reasons, the program was written directly in Pascal. This afforded an opportunity to determine the degree to which a structured general purpose language such as Pascal is suitable for the presentation of tutorial lessons, and in particular how it performs when compared to the specialized tutorial authoring languages such as MASTER. The program was constructed so that, as when using a specialized tutorial language, the text to be displayed may be read from separate disk files, but by contrast with specialized language use the presentation of the text is controlled from within the program, as is all the other graphical display.

Questions are asked during the course of the lesson to determine whether or not the most elementary material should be displayed and whether further explanation is required for

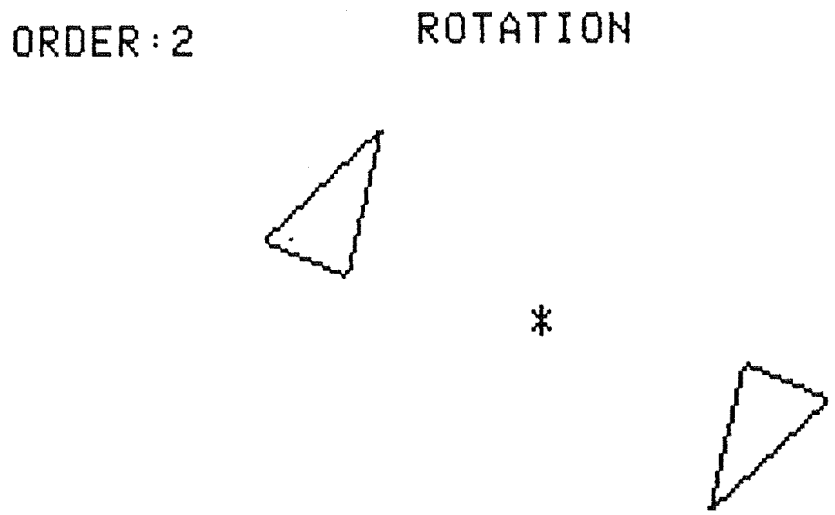


Figure 10.7 Sample display from the tutorial program TUTORIAL.

a particular demonstration. The routine used to analyse the student responses in this Pascal program was taken directly from the MASTER interpreter, with the only changes being the manner in which the student's responses are passed to the procedure. All responses transmitted to the answer matching routines were saved on a disk file for later analysis by the teacher.

The first demonstrations given by the program are of non-translational operations encountered in point groups (reflection, rotation and inversion) and may be bypassed if the student has had previous experience in this subject area. Demonstrations of the glide and screw operations are the next to be presented, introducing the concept of translation as a symmetry operation. The short discourse on handedness which follows marks the end of the first half of the lesson in which general concepts only are considered.

The second half of the lesson deals with specific algebraic relationships between objects related by symmetry operations. The demonstrations of reflection, two-fold rotation and the glide operation are repeated with descriptions of how the coordinates of an object at an arbitrary position are altered by the application of the operation currently being displayed. The two-fold screw operation is also examined although no animated demonstration is given. The final demonstrations are of three fold-rotation and screw axes which includes examination of the effect of repeated applications of these operations.

Comparison between program TUTORIAL and lessons written in specialized teaching languages, or programs capable of being used as auto-elaborative teaching aids leads to the conclusion that while a tutorial program can be written successfully in a general purpose programming language it is not as suitable for teaching use as the alternatives. In particular, control over the direction the lesson is going is greatly facilitated with a specialized language with the matching of responses and the subsequent branching being the most significant factor in this. Another important consideration is the time taken to develop a tutorial, which in a specialized language such as MASTER is much less than the time necessary to develop a similar lesson in a general purpose language such as Pascal. However the one important advantage that a general purpose language has is that a greater variety of graphics types is available. This becomes significant when a particular effect, such as animation, is necessary for the most effective presentation of ideas in a lesson.

10.3 THE SYMOP LESSON

As a follow-up to the use of either SYMPACK or TUTORIAL a drill and practice revision lesson was prepared. This lesson, named SYMOP, tests the student by asking questions based on diagrams generated by program SYMPACK. Its function is thus analogous to the role of the lessons, described in chapter 9, which accompany program SGROUP. As

with the SGROUP lessons, SYMOP is written in the MASTER tutorial language and is included in the software package described in section 5 of appendix A.

SYMOP begins with a brief page of instructions including a description of the axial conventions to be used, which are the same as those used in other programs in the package. This is then followed by eleven questions based on a series of diagrams originally generated by SYMPACK but with some modifications (see figure 10.8). The questions are mainly concerned with the algebraic relationship between objects related by the operation currently being displayed, although in the case of reflection, rotation and screw there are also more general questions on the effect of the operation. As well as the questions on the algebra of each of the operations covered in SYMPACK there are also questions relating to the effect of the application of a second operation to an object. The first of these covers the effect of adding a mirror plane parallel to a rotation axis, while the second deals with a mirror plane perpendicular to the direction of translation in a glide plane. In these cases the emphasis is as much on the resulting operation as the algebraic outcome.

10.4 POINT GROUPS

Program PGROUP was designed to demonstrate standard point group diagrams in a fashion analogous to the way SGROUP displays space group diagrams [20]. The diagrams

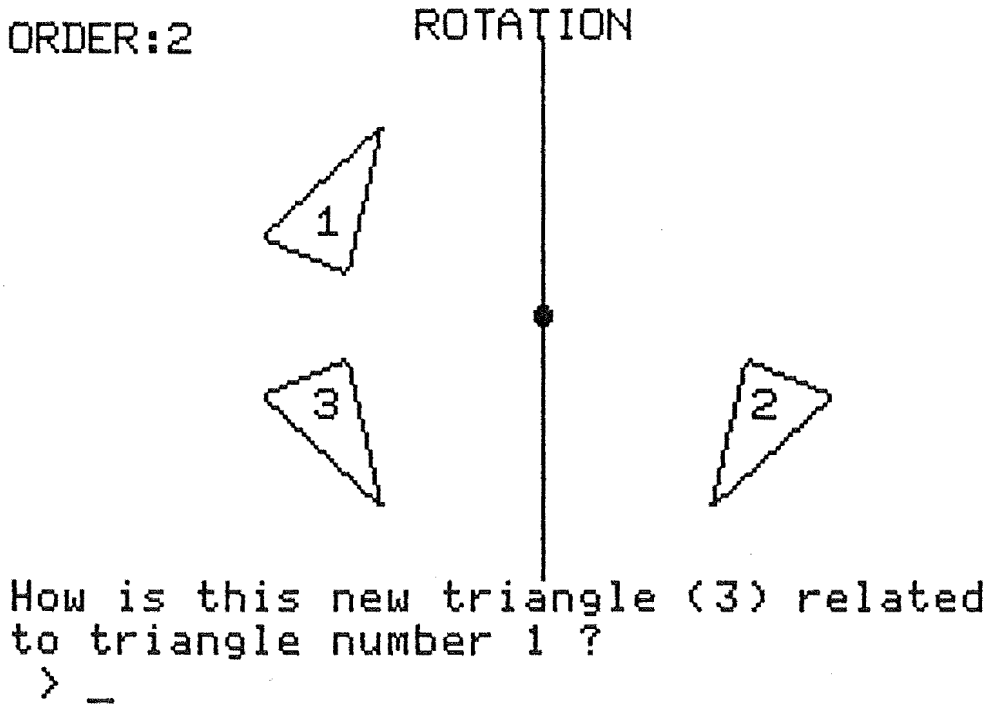


Figure 10.8 Sample display from MASTER lesson SYMOP.

generated by PGROUP are stereographic projections of all the non-cubic point groups (see figure 10.9). The symbol of a point group in either Hermann-Mauguin or Schoenflies notation is first entered. The diagram for this point group is then generated by the program, after which other symmetry in the form of two-fold rotations, reflections, or a centre of inversion may be added or deleted from the display. The principal axis however cannot be directly modified unless a new diagram is generated by entering the appropriate symbol. A symmetry operation may be deleted from a diagram but the symbols for it will be regenerated automatically if it is implicit in other operations already present. However when new operations are added the symbols for implied operations are not generated automatically and must be added by the user, although the correct point group symbol is displayed even if all the operation symbols are not present.

Written in Pascal and forming part of the Symmetry Operations package, PGROUP was written primarily as a lecture demonstration tool although as with most such programs in this project it has also found use on an individual user basis. Point groups are first introduced to students in the honours part I class where PGROUP receives greatest use as a lecture demonstration aid, but it is also used in the honours part II class for a revision of point groups as an introduction to space groups.

The program enables the teacher to quickly call up any point group diagram and then modify it. This is most usefully done by generating only a principal axis then

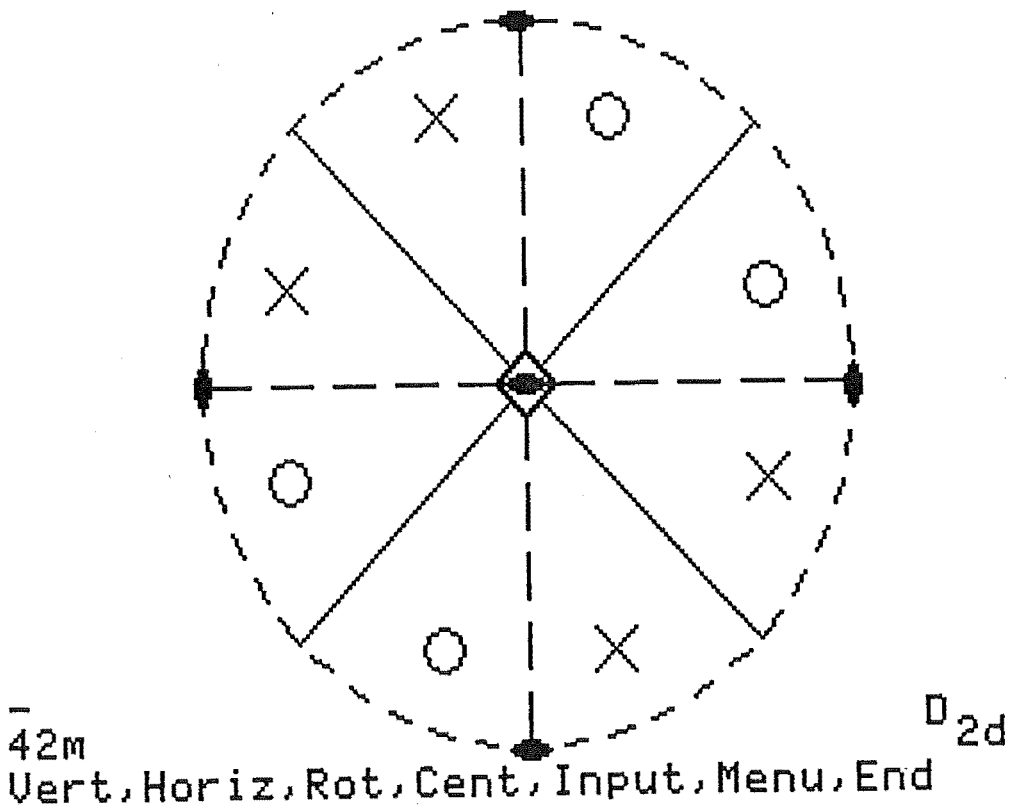


Figure 10.9 Sample display from program PGROUP. The Hermann-Mauguin and Schoenflies symbols for the displayed point group are given to the left and right of the display respectively, immediately above the command prompt line.

building up a more complex point group in a stepwise fashion analogous to the generation of space group diagrams in SGROUP. PGROUP may also be used to show how a third operation may be generated by the combination of two other operations. This may be achieved by simply adding two operations to a diagram and noting the effects, then adding the third (implied) operation and noting its effect.

Both the honours part I and part II classes used the program on an individual basis as a backup to the material presented in lectures. Typical student activity consisted of building up diagrams of the more complex members of each class of point group by starting with only the principal axis of that class (see figure 10.10). Particular emphasis was placed by the students on the correspondence between the operations added and the change in point group symbol which occurred when the diagram was altered.

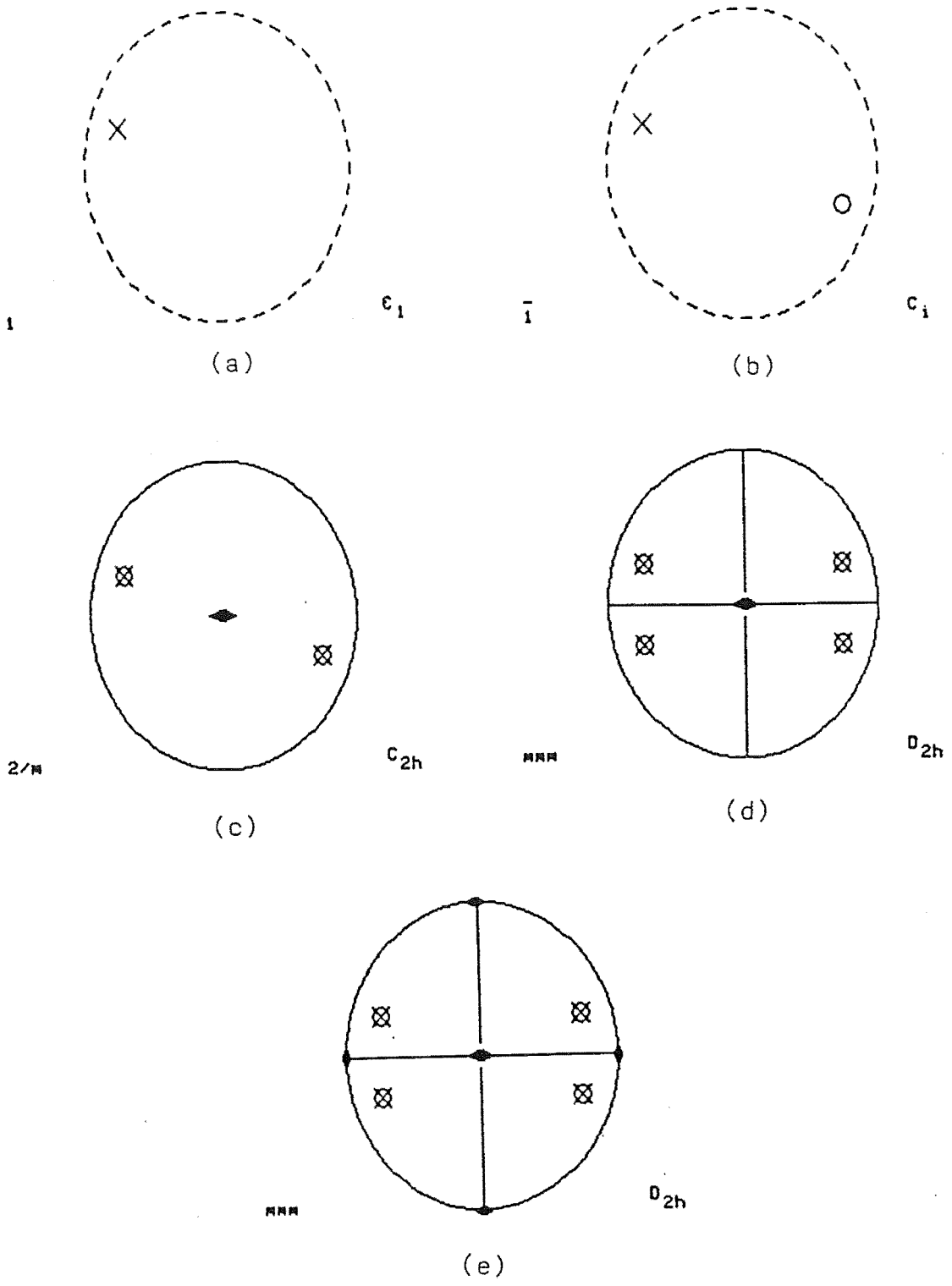


Figure 10.10 Steps during the building of a point group diagram. a: The initial diagram. b: A centre of inversion is added. c: A horizontal mirror plane is added. The implied two fold rotation is automatically added by the program. d: Vertical mirror planes are added. e: The implied horizontal rotation axes are added to complete the diagram.

CHAPTER XI

CRYSTAL SYMMETRY AND CRYSTALLOGRAPHY

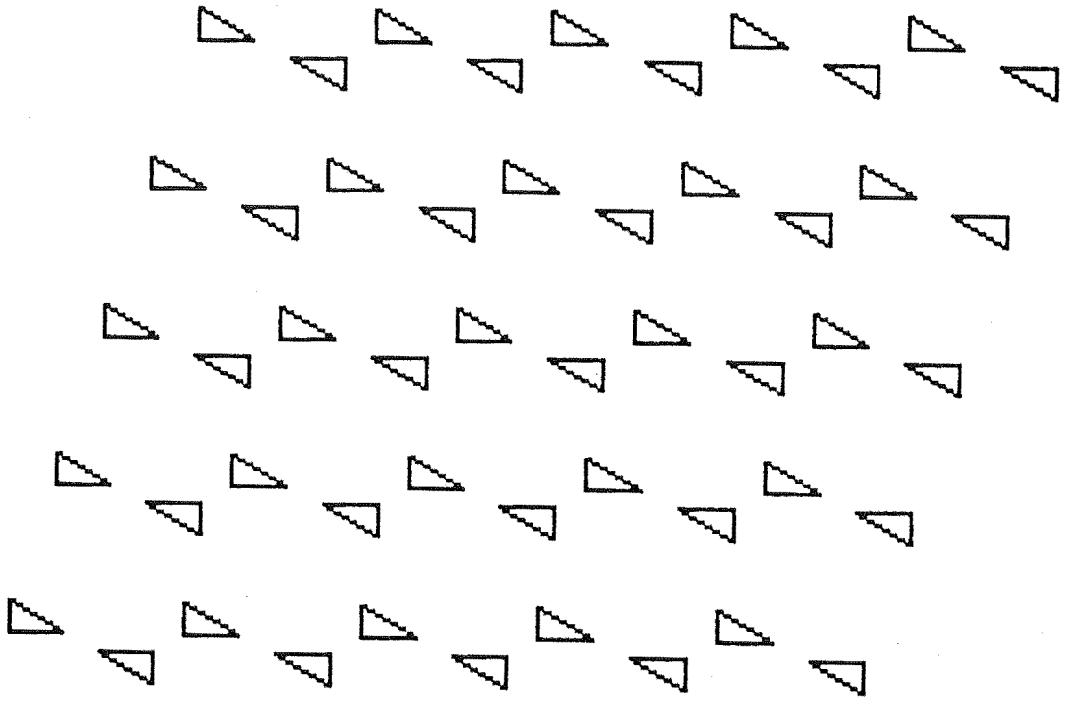
Central to the understanding of three dimensional crystalline structure is an appreciation of the range of permitted crystal lattices and the manner in which molecules or molecular fragments may be packed in them. With this in mind a number of lecture demonstration programs were written to pictorially illustrate, in simplified form, some of the elementary characteristics of crystal lattices. The areas covered in these programs include the arrangement of objects in two dimensions, the choice of unit cells in two dimensions, the types of three dimensional unit cells, the labelling of planes within a lattice and the relationship between lattice planes and incident X-rays. All programs were written in Pascal and with one exception were included in the symmetry Operations Package

11.1 CRYSTAL ORGANIZATION

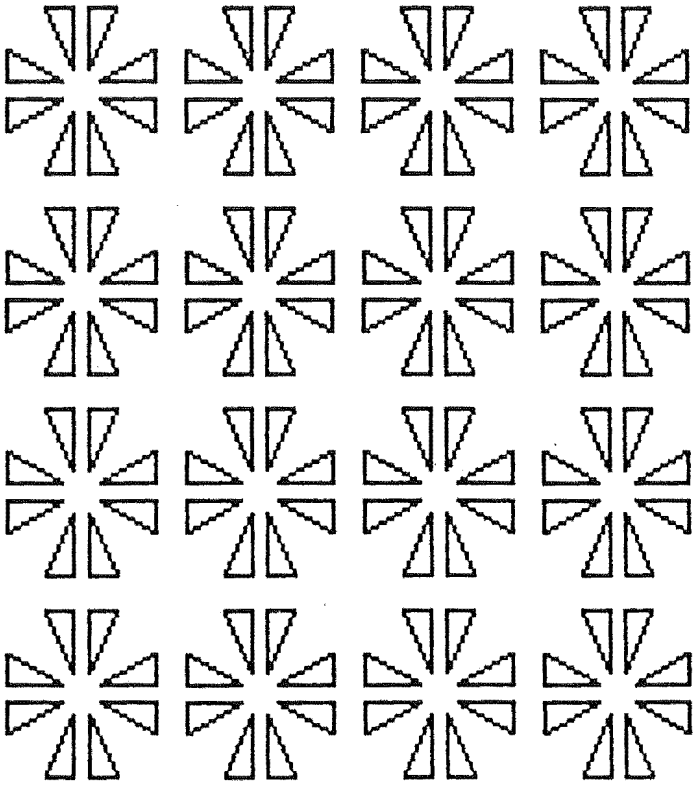
The "Plane Patterns" program (file name SEVENTEEN) is a demonstration program intended to illustrate the idea of crystal symmetry as distinct from molecular symmetry. This is achieved by the generation of regular patterns made up by the repetition of an asymmetric object on a two dimensional surface [54]. This asymmetric object, a scalene triangle, may be replicated to form any one of the seventeen possible

plane lattices by entering the appropriate parameters as they are requested by the program (see figure 11.1). The first parameter is always the type of lattice, which may be parallelogram, rectangular, square, or hexagonal. The following parameters relate to rotations, glides and mirror planes which may be present. The requests for these parameters vary according to what symmetry is possible with each lattice type. Once all parameters have been entered the display is generated. A second display may be superimposed on the one already generated to enable the teacher to build up a series of diagrams containing increasing symmetry. This is only possible if the lattice type is unchanged, otherwise the existing display is erased to accommodate the new display.

The choice of unit cells in a two dimensional lattice is the subject of the "Unit Cells" program, which uses the filename TWOD. During operation the program displays a two dimensional grid, to which a flashing unit cell outline may be added (see figure 11.2). A choice of unit cells is given to illustrate that more than one unit cell is possible for any given system. When using the program, the desired grid system must first be chosen from the a menu which covers the full range of systems which are possible in two dimensions. Once the grid has been generated a prompt line is displayed which specifies the keys to be pressed to generate the available unit cells. The unit cell outlines flash in different colours for added distinctiveness although the program may be used satisfactorily with a monochrome



(a)



(b)

Figure 11.1 Sample displays generated by the "Plane Patterns" program. a: Pattern p2, a parallelogram lattice with two fold rotation. b: Pattern p4mm, a square lattice incorporating mirror lines.

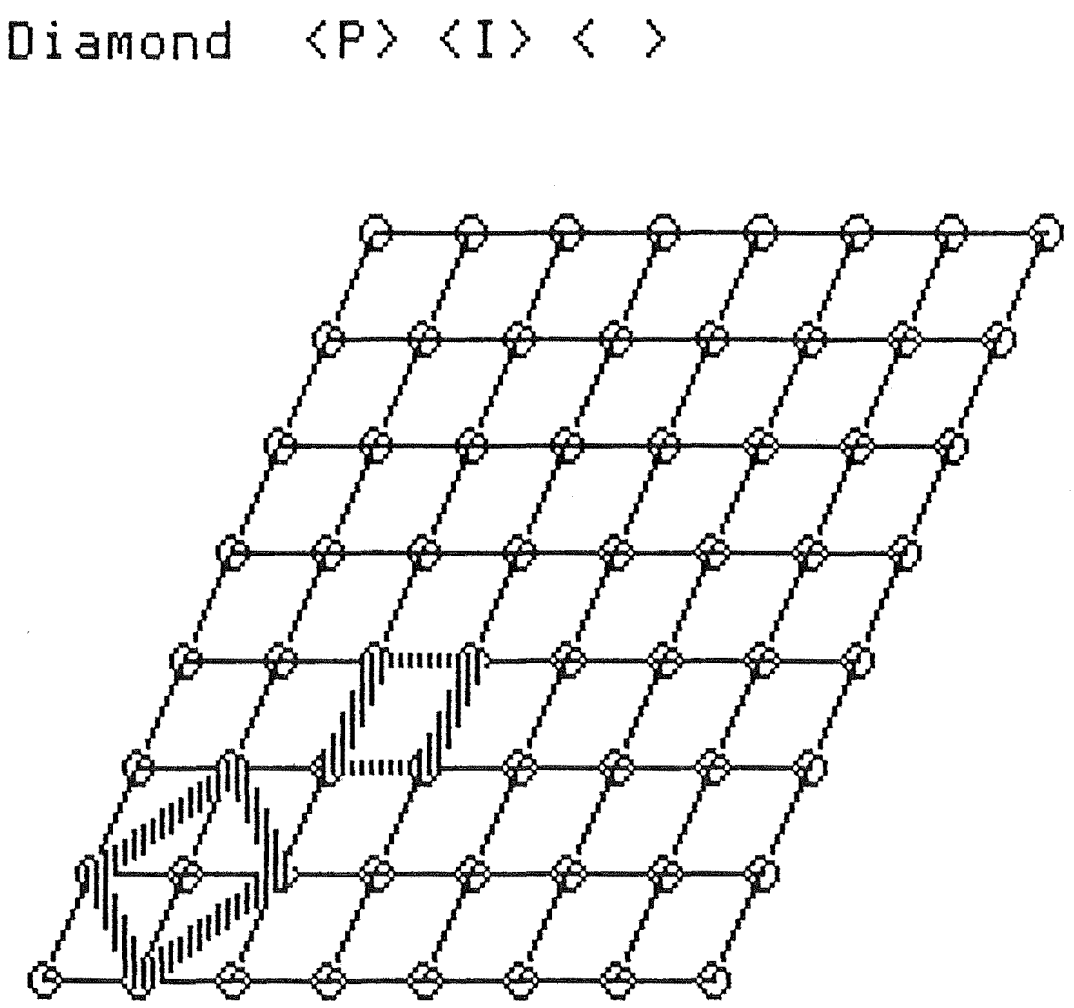


Figure 11.2 Sample display from the "Unit Cells" program. A primitive and centered unit cell in a diamond lattice are highlighted.

monitor.

The step from illustration of the two dimensional to the three dimensional lattice system involves the description of the types of lattices which are possible. Diagrams of the types of unit cell possible in these lattices are generated by the "Bravais Lattices" program. This program, which uses the filename LATTYPES, first generates a diagram of one of the possible general types of system as specified by the angles between axes and relative cell edge lengths (see figure 11.3). The diagram is of the form of a perspective line diagram with circles at the vertices and is accompanied by a line of text giving the algebraic description of the system. The system to be displayed is selected by the teacher from a menu in the same manner as in program TWOD. Lattice centring may then be added to the diagrams in the form of circles at appropriate locations with a promptline giving the types available in each diagram (see figure 11.4).

The program not included in the Symmetry Operations Package deals with the close packing of spheres as used as a structural model for metallic solids. The program was not included in the package since disk space was limited and the material dealt with in the program was of a more elementary nature than that in the other programs. The packing effect is illustrated in the program (filename PACKING) by layers of circles which may be superimposed on each other to represent the effect of a series of layers of spheres (see figure 11.5). Colour is used to differentiate the different

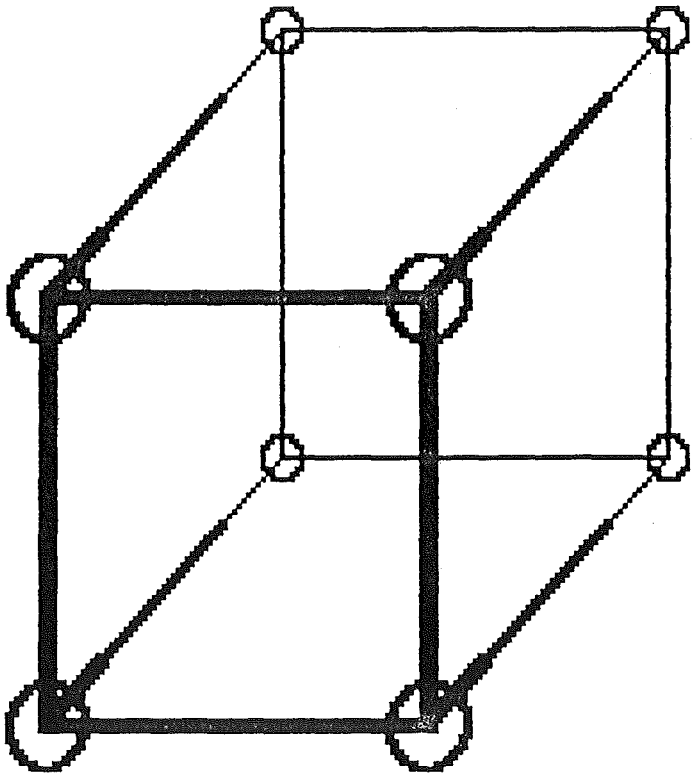


Figure 11.3 Tetragonal unit cell type as displayed by the "Bravais Lattices" program.

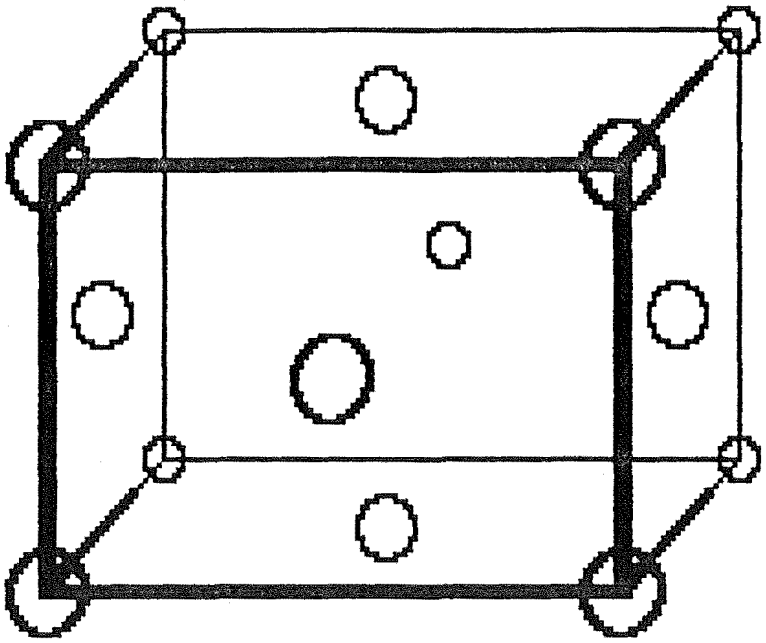


Figure 11.4 Orthorhombic unit cell incorporating face centring as displayed by the "Bravais Lattices" program.

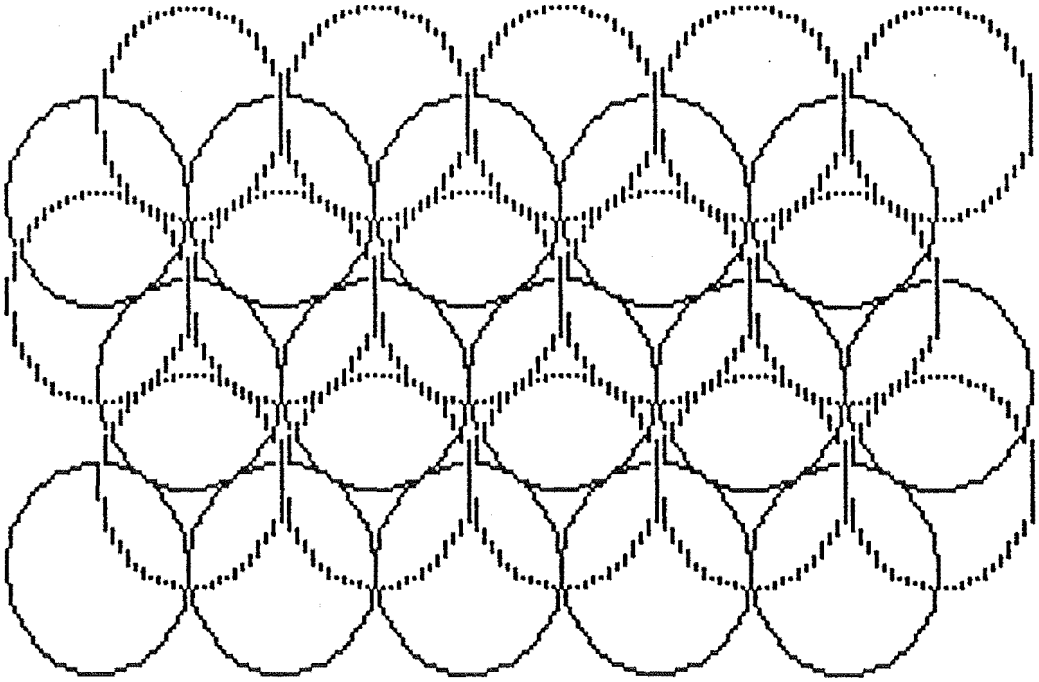


Figure 11.5 Sample display from program PACKING. Shown are the first two layers of a packing diagram. On a suitable video monitor the layers will appear in differing colours.

layers, and circles of slightly diminished radius are also possible so that the "lower" of two directly superimposed layers remains visible. Circles of significantly reduced size may also be added to represent atoms in the interstices within the lattice. All control in this program is by single keystrokes given by a promptline of valid commands.

The labelling of crystal lattice planes by Miller indices is a simple device of central importance in crystallography, but one which students traditionally find difficult. Program MILLER was written as an aid to its understanding. The program uses a non-orthogonal two dimensional grid on which may be superimposed lines corresponding to the Miller indices entered by the user (see figure 11.6). The possible magnitude of the indices ranges from zero to nine with negative indices also being allowed. More than one set of lines may be drawn at once onto the grid with second and subsequent lines being coloured to ensure that they may be differentiated.

The accepted qualitative basis for the explanation of x-ray diffraction by crystals is that x-rays of a fixed wavelength appear to be reflected from a particular set of lattice planes when the Bragg condition is fulfilled. One way of illustrating this idea is to show that the resulting waves from a waveform partially reflected from a series of parallel planes of fixed spacing will only constructively interfere at a particular incident angle. In program BRAGG, a waveform is generated on screen where it is reflected from three parallel planes with the angle of incidence controlled

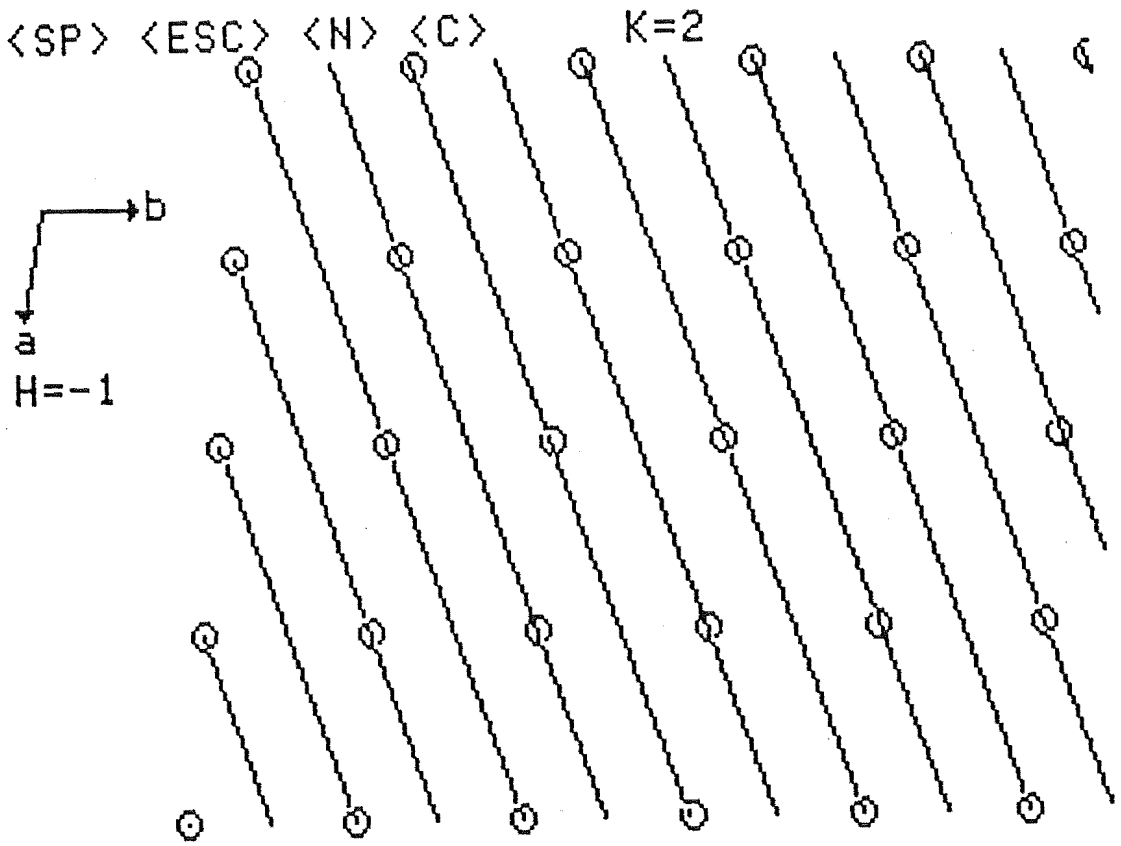


Figure 11.6 Sample display for program MILLER.

by the user (see figure 11.7). The teaching strategy associated with this program is to try various incident angles until the angle at which the reflected waves match is found. A flashing line perpendicular to the reflected waves is provided to determine if they do coincide with one another. A facility to control the colour of the waves is provided so that waves at more than one angle may be shown.

11.2 PROGRAM USE

The programs described in this chapter are fundamentally lecture demonstration programs, but unlike other such programs examined in earlier chapters these do not have a secondary role in individual student use except when a student wants to simply repeat the material presented through the program in a lecture. The distinction arises because of the difference in the number of pieces of information which are transmitted through the different types of program. Longer programs, such as SGROUP and PATTERSON, have the potential to show many aspects of the subject they are designed to help explain, and this is facilitated by the various optional display modifying commands available to the user. Programs with less possible variation in display, such as those described in this chapter, are consequently much more limited in scope so that everything covered by the program may be presented in a relatively short period of time during the lecture.

Since individual use of these programs was confined to

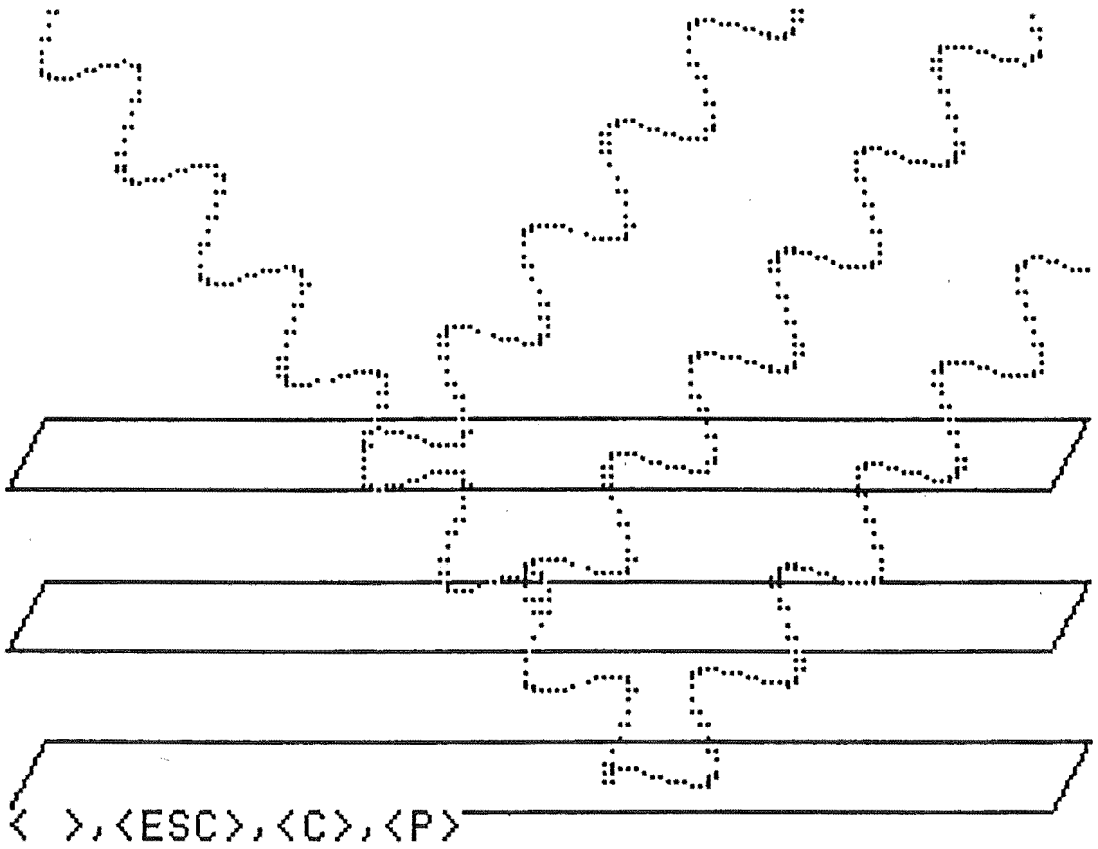


Figure 11.7 Sample display for program BRAGG. The angle of incidence for the wave form is 57°.

revision sessions the only supplementary material provided for students were instructions which enabled the lecture demonstrations to be repeated. Those programs which had printer dump facilities could also be used to generate diagrams for later reference.

While lacking the auto-elaborative possibilities of the more complex demonstration programs, the programs described in this chapter still retain advantages over other forms of demonstration in the classroom or lecture theatre. Programs SEVENTEEN, LATTYPES and TWOD are all constructed so that they may display any one of a series of diagrams corresponding to the different states possible in the system being examined. Displays of this type may also be accomplished using slides or transparencies but these systems lack the facility to easily add additional features to the current diagram, as well as being less flexible if alterations in display order are required.

Where the appearance of the quantity to be displayed is dependent on parameters which can vary over a large number of possible values it becomes effectively impossible to show this variation adequately by a series of pre-prepared diagrams. However in programs such as MILLER and BRAGG, any one of a large number of potential displays can be generated through the use of a general computer algorithm. This gives a flexibility of display not possible using other techniques.

CHAPTER XII

PROGRAMMING DETAILS

The scope of any software development will always be limited by the capabilities both of the computer and of the languages used. To achieve aims when writing a program it is often necessary to develop special utilities or routines rather than depend on the direct use of the standard features of the computer or language. In other cases it may be the manner in which standard features are employed which produces the desired result. The programming techniques developed during this project deal mainly with graphics generation, character string and file handling, animation and methods to overcome the shortage of peripheral storage and program memory in the computer.

12.1 GENERAL PROGRAMMING CONSIDERATIONS

The provision in the Apple Pascal system for segmentation of programs means that when a program becomes too large to be accommodated entirely within program memory, certain parts of the program may be designated as segments, which are only loaded into memory from disk storage when they are actually called during program execution [61]. Although this provision does increase possible program size it also has the potential to significantly slow program execution speed as there is a delay every time a segment is

called. It was therefore necessary to avoid the segmentation of procedures which would be called often. In programs requiring extensive initialization, in particular SGROUP, PATTERSON and PGROUP, the routines responsible for this initialization could be segmented as they would only be called once at the beginning of the program. Other infrequently called routines which could be segmented include graphics bit image dumps to a printer, sets of instructions for the user, and for PATTERSON and SGROUP, orientation changing commands and unit cell information reading procedures could also be segmented. Programs SYMOP and TUTORIAL present a different problem as, while they are large programs they do not have many individual routines suitable for segmenting. Therefore the sections corresponding to each of the demonstrations in the programs were constructed as separate segments.

Although not used during program execution the 80 column display was used during program development as were "macros" designed to reduce the number of keystrokes needed to enter commonly used programming commands when using the program text editor. Other software used during program development was concerned with the automatic setting up of the 128K RAM card as a virtual disk, and the use of the clock card to ensure the system had the correct date for the labelling of files. The software for using a lower case chip is also included in this category.

12.2 GRAPHICS TECHNIQUES

In order to develop smooth animation for the programs demonstrating symmetry operations, it was necessary to use a two page graphics system. In this system flicker free animation may be created by displaying one graphics page while the next picture in the animation sequence is being constructed on the page not being displayed. The animation progresses by swapping the roles of the two pages. There is no provision for such a facility in standard Apple Pascal but there is in Applesoft. It was therefore implemented by constructing utilities which mimicked the relevant Applesoft commands [62]. The provision of a second graphics page effectively reduced the available program memory, and could be used only when an 80 column card was not present or was deactivated. It does however constitute the most satisfactory method of animation on the Apple computer.

Graphics for the MASTER language is provided in the form of complete pictures loaded from disk files. Initially the disk files were created by direct transfers from computer to disk of the graphics section of memory, but a second superior system involving compressed picture files became available later. The compressed files were eventually used for all MASTER programs during the course of this project as they loaded faster, had lower memory overheads on loading, could perform boolean operations with the existing display, and required less disk storage space. The utility programs PICCY and SHOW are used to create and

manipulate picture files for use in MASTER programs or for use as diagrams which may be sent to a printer. Most pictures used in the MASTER programs are originally generated by one of the lecture demonstration programs and saved to disk using PICCY or SHOW. If any modifications have to be made to the picture, they are made using PICCY which has provision for adding lines, boxes, circles or lines of text to the picture. Program SHOW is used in cases where pictures must be merged, where the picture is on the second graphics page or where the picture is to complex too be compressed using PICCY. If space must be made to accommodate the text part of a lesson, a picture may be moved vertically on the screen by use of program SCROLLER.

The character set used for text display on the graphics screen has been modified to allow customized characters to be used in programs. The programs in which the characters are used include PATTERSON, LATTYPES, MILLER, PGROUP and all MASTER lessons. The types of special characters used include greek letters, mathematical symbols, subscripts, fractions and other special pictorial shapes. Program CHED is used to modify the character set, which it does by giving an expanded view of a character which may be edited and saved back to disk (see figure 12.1). An enhanced version of CHED was developed which allowed for the transfer of characters between different character sets.

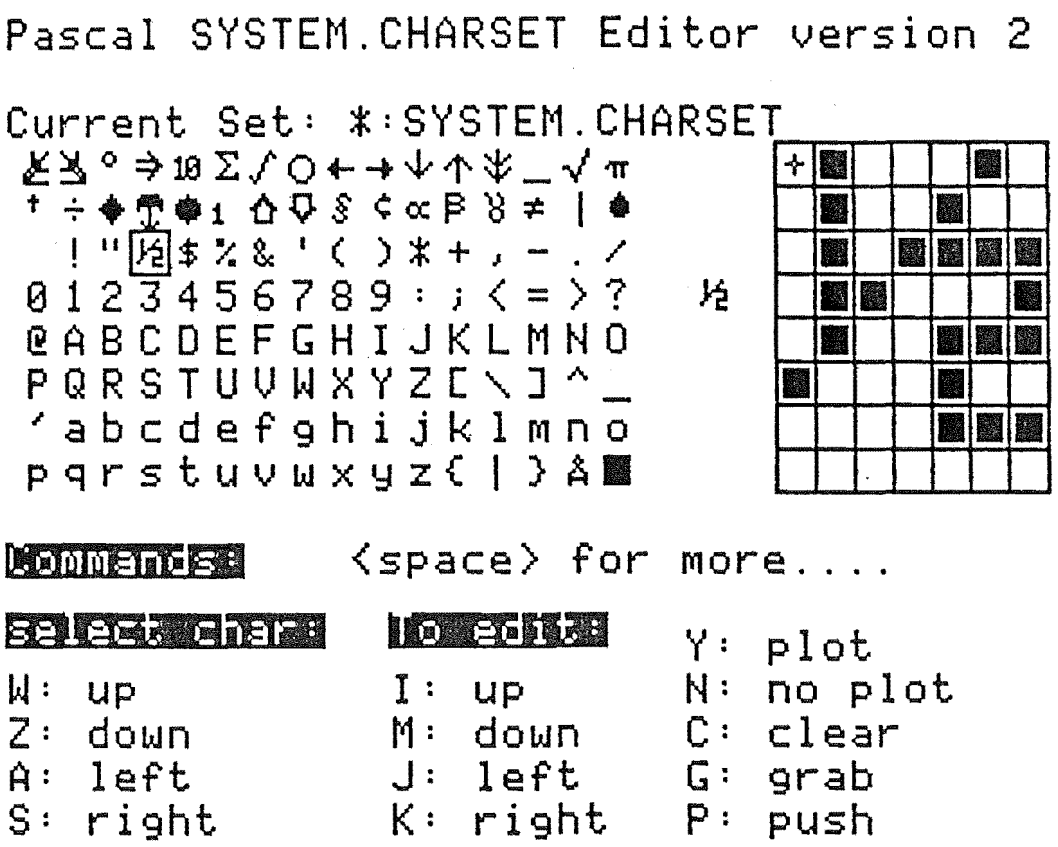


Figure 12.1 Sample display for program CHED. The character set displayed is the one developed for use with programs in this project.

12.3 INPUT/OUTPUT

The first major problem experienced with Apple Pascal was the unsuitability of the standard numeric input routines for use by inexperienced users. Any mistake made during the input of real numbers causes the program to terminate immediately. The solution to this problem was to treat the numbers as character strings as such strings can be more readily adapted to any desired mode of use than can numerical variables. To make use of numbers entered in this way it was necessary to determine the value of the number entered so that it could be transferred to numeric variables for use in calculations. Two routines were written to perform this function. The first was a general procedure for extracting several numbers from a string, while the second was of a simpler form for those programs where only one number would be entered at any one time. The use of strings also allowed the input to be displayed on the graphics screen as it was entered.

Not all input however is in the form of strings, in most programs the commands are entered as characters which are selected from a menu. Such a system allows for rapid selection of commands and no overheads for string matching routines or display of commands on the graphics screen as they are entered. Since the nature of the commands varies it is not feasible to have a standard set of commands for all programs. However it is possible to have conventions for those commands which do appear regularly. In particular

the commands for leaving the program, producing a printer dump, clearing the screen and displaying menus of commands could be standardized.

Programs PATTERSON and SGROUP have provision for the reading of molecule information files from disk, but the methods used differ considerably. When PATTERSON was first written, provision was made for both reading from and writing to real number data files on disk. Such files are compact and easy to set up, but their manipulation is confined to programs specifically designed for them, such as PATTERSON.

SGROUP was not originally designed with any provision for access to disk files, but when it was decided to add the facility to read files from disk, lack of program storage space in the computer meant that it was not possible to implement a complete new set of procedures such as those in PATTERSON. Instead those sections of the program used to read molecule data from the keyboard were modified so that input could also be taken from text files containing strings in the same format as they would be entered from the keyboard. No provision was made for saving such text files from within the program, but they could be created using the Pascal system's text editor. Text files have significant disk space overheads which make them inefficient for the storage of small files such as the molecule files, so program PACKER was written to create more compact data files containing the same information. Once SGROUP had been modified to accommodate either type of file, it was possible

to use the more compact files in packages where conservation of disk space was important. Eventually program MOLEDIT was written which allowed for the direct construction and modification of the compact molecule data files without recourse to the text editor and PACKER.

The loading of disk files, such as those used in SGROUP, normally involves the automatic creation of a buffer area which reduces the amount of memory available for the actual program. To make more space available for SGROUP, a file loading system was devised which moved the file buffer to an area of memory not normally available to a program. Although generating more program code this technique did make available more useable memory and had the extra advantage of being faster than standard file loading techniques. This technique may only be used for text files or data files made up of text strings, so it is not suitable for program PATTERSON which uses real number data files. Conversion of PATTERSON to a text based file format offers no significant advantage as the standard type of file buffer must still be generated for the loading of files containing shape information for graphics. These shape files, which are also used in PGROUP and SYMPACK, cannot be converted to a text based format.

CHAPTER XIII

ANALYSIS OF STUDENT SURVEYS

To assess the effectiveness of software developed during the course of this project the main target groups of students were asked to complete questionnaires relating to the subject area and the use of computers. The honours part II class was the group receiving the largest section of course content devoted to the study of symmetry concepts and was eventually the group exposed to the largest proportion of the programs contained in the symmetry operations package and related revision lessons. The classes for the years 1981, 1983 and 1984 were examined separately since they had each received significantly differing amounts of instruction through computers.

13.1 SURVEY DETAILS

The 1981 class was exposed to computer based teaching techniques as part of the course on symmetry and crystallography for only a single 50 minute lecture period. The class was therefore suitable as a control group and it was also used to supply information as to the most appropriate topic areas for subsequent program development.

The survey administered to this group consisted of a list of 43 concepts relating to crystallography which the students were asked to evaluate according to the ease with

which each concept was understood (see figure 13.1). The survey was conducted following the completion of the appropriate lecture course.

Computer based teaching was used during lectures for the 1983 class in the fields of point groups, space groups, Miller indices and two dimensional plane patterns. Members of this class were given the opportunity to use the computer on an individual basis to examine properties of space groups using program SGROUP as well as attempting the space group and unit cell revision lessons. This class was asked to complete the same concept difficulty survey as the 1981 class plus a second survey relating to time spent using the computer and any perceived benefits from such use (see figure 13.2). Because several weeks elapsed after the last student had used any of the teaching programs on the computer there was a considerable time lag for the students between the final contact with the subject matter and the administration of the survey.

For the 1984 class programs were added to deal with lattice systems and Bragg reflection, there was increased use of SGROUP and revision lessons were upgraded. In addition to the concept difficulty survey the 1984 class was also asked to complete attitude surveys both before and after the course [63]. The first of these attitude surveys, which was given at the beginning of the lecture course, contained nine items relating to familiarity with computers and expectations of the usefulness of computers and their relationship to the subject area (see figure 13.3). The

SURVEY OF CONCEPT DIFFICULTY IN CRYSTALLOGRAPHY

Indicate, by using numbers 1 to 5, the degree of difficulty you encountered in understanding the following concepts (1 = no problems, 5 = great difficulty).

C1	Regularly repeating 3-D arrays of points with identical environments	____
C2	Lattices formed from such arrays	____
C3	Primitive unit cells as representative volume elements of a lattice	____
C4	The characteristic symmetry of the 7 crystal systems	____
C5	Centered unit cells (A, B, C, F and I)	____
C6	7 crystal systems + 14 Bravais Lattices	____
C7	Crystallographic axes (non-orthogonal & orthogonal) appropriate to the seven crystal systems	____
C8	Unit cell edges as units of length along each of 3 crystallographic axes	____
C9	Fractional coordinates defining general atomic positions with respect to crystallographic axes	____
C10	Symmetry operations about a point	____
C11	Schoenflies and Hermann-Mauguin symbols	____
C12	Alternating and inversion improper symmetry axes	____
C13	Collections of symmetry elements as point groups	____
C14	Point group derivation for discrete objects (molecules)	____
C15	Stereographic projections of point group symmetry elements	____
C16	Unit cell translation as a symmetry element	____
C17	Screw axes of any order	____
C18	Glide planes and their labelling	____
C19	Indications of symmetry elements involving translation by systematic absences of X-ray reflections	____
C20	Graphical representations of symmetry elements	____

Figure 13.1a Page 1 of the concept difficulty survey. Items C1 to C20.

C21	Space groups as combinations of symmetry elements with lattice types	____
C22	Space group labels (e.g. $P2_1/c$)	____
C23	Space group symmetry element diagrams	____
C24	Space group equivalent position diagrams	____
C25	Algebraic representation of equivalent positions	____
C26	Special positions and their point symmetries	____
C27	Asymmetric units	____
C28	Atoms with respect to general positions in unit cells	____
C29	Molecular symmetry information from space group data	____
C30	Atomic scattering of X-rays	____
C31	Stacks of lattice planes with constant interplanar distance d	____
C32	Miller indices (+ve and -ve) as identification for lattice planes	____
C33	Diffraction represented as reflection	____
C34	Bragg's law	____
C35	Diffraction of X-rays by crystalline powders	____
C36	Analysis of X-ray powder diffraction patterns	____
C37	Identification using X-ray powder data file	____
C38	Determination of cell constants from an X-ray powder pattern	____
C39	Measurement of single crystal X-ray diffraction intensities	____
C40	Structure factor derivation from intensity observations	____
C41	Structure factor calculation from atom positions	____
C42	Fourier summation and electron density	____
C43	Heavy atom phase determination	____

Figure 13.1b Page 2 of the concept difficulty survey. Items C21 to C43.

Survey of Attitudes to Computers in the Teaching of Symmetry Concepts

Please circle the response which most closely reflects your opinion.

Lecture demonstrations.
Please rate the effect of the computer based lecture demonstrations using the following scale. 1=very unclear....5=most effective.

A1	Point groups	1	2	3	4	5
A2	Space groups diagrams	1	2	3	4	5
A3	Two dimensional plane patterns	1	2	3	4	5
A4	Miller indices	1	2	3	4	5

Individual use of the apple computer.
Rate the effectivness of the following computer tutorial sessions. 1=not worthwhile....5=very useful. If you did not use a section of these programs then do not give them a rating.

A5	Point group revision	1	2	3	4	5
A6	Space group generation	1	2	3	4	5
A7	Space group questions	1	2	3	4	5
A8	Unit cell questions	1	2	3	4	5

Evaluate the following factors on the scale
1=very unsatisfactory...5=very good

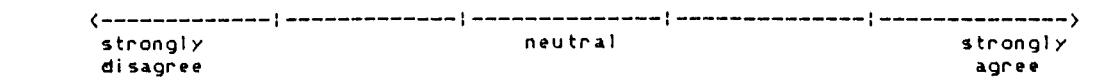
A9	The degree of computer availability.	1	2	3	4	5
A10	The introduction to use of the computer.	1	2	3	4	5
A11	The ease of use of the programs.	1	2	3	4	5
A12	The correspondence between the course content and the computer material	1	2	3	4	5
A13	Completeness of coverage (were there gaps in the material?)	1	2	3	4	5

If you have any comments relating to the computer assisted learning aspects of the course, please enter them in the space below.

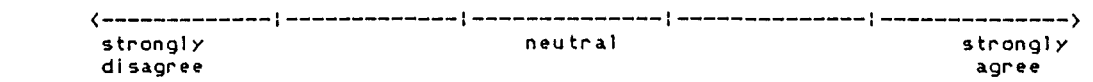
Figure 13.2 Attitude survey (1983) taken subsequent to computer based course.

Please read each of the following statements and record your reaction on the scale below the question.

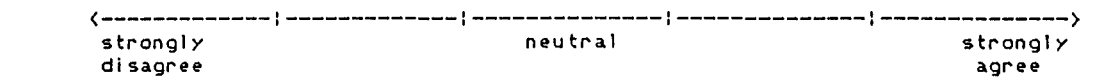
P1 I am very familiar with computer use.



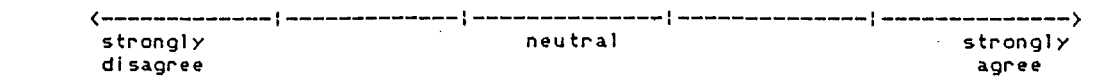
P2 I do not consider the field of symmetry to be an important part of chemistry.



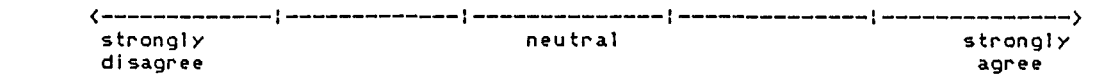
P3 Computers will enhance my ability to comprehend crystal symmetry.



P4 It will be difficult to take notes from computers used in lectures.

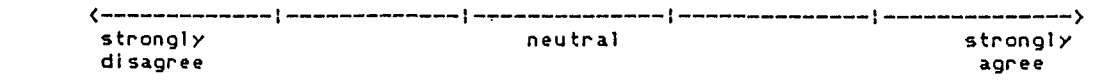


P5 Using computers by myself will be more useful than in lecture demonstrations

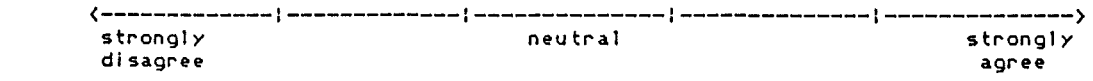


P6 Computers in education is just a gimmick.

P7 I am very eager to use computers to help me learn this subject.



P8 Symmetry is not a suitable field for computer assisted learning.



P9 To date I have had no trouble understanding symmetry in chemistry.

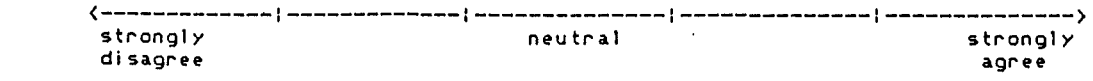


Figure 13.3 Preliminary attitude survey (1984). Items P1 to P9.

second attitude survey, containing nineteen items, which was given to the students following the course, re-asked the questions contained in the first survey so that any changes in attitudes would be noticed. Some further more specific questions about the use of computers during the course were also asked (see figure 13.4). In both attitude surveys each item in the survey covered a single idea or attitude to avoid confusion and the statements comprising the survey were both positive and negative in nature so that the students were not influenced by any perceived expectations. The concept difficulty survey was given at the same time as the second attitude survey although six items were deleted as the concepts involved were not covered in the course material for 1984. It was possible to conduct the later surveys at the stage when most students had just completed all the individual exercises on the computer thereby avoiding the delay which occurred before the administration of the 1983 survey.

13.2 CONCEPT DIFFICULTY RESULTS

A major difficulty encountered when attempting to draw conclusions from the results obtained from these surveys is that the small sample sizes involved mean that genuine trends must show very large effects to have any statistical validity. The numbers of students involved were twelve, fifteen and thirteen for the 1981, 1983 and 1984 surveys respectively. Means and standard deviations for the concept

This survey is designed to measure your attitudes towards the use of computers in the teaching of symmetry concepts. The results from the survey are to be used as part of a PhD. research project, and for planning future computer related teaching projects. Please answer each section as accurately as possible to ensure a valid result.

(Note: In the following survey the '"STAF" tutorial lessons' refers to the revision programs to be found on the "STAF" disks - It does not refer to the set of exercises which accompany the space group program.)

T Please indicate the number of hours that you have spent using the computer for each of the following.

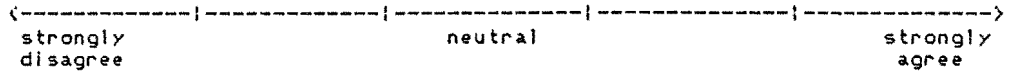
The space group program. ____

"STAF" tutorial lessons. ____

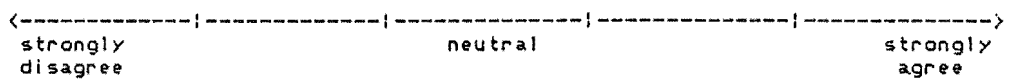
Other related programs ____

Please read carefully each of the following statements and record your reaction on the scale below the question.

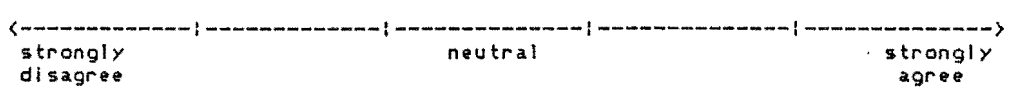
S1 I am familiar with computer use.



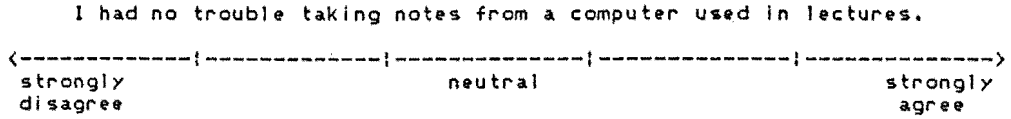
S2 I do not consider the field of symmetry to be an important part of chemistry.



S3 Computer use has enhanced my comprehension of crystal symmetry.



S4 I had no trouble taking notes from a computer used in lectures.



S5 Using a computer by myself has been more useful than in lecture demonstrations

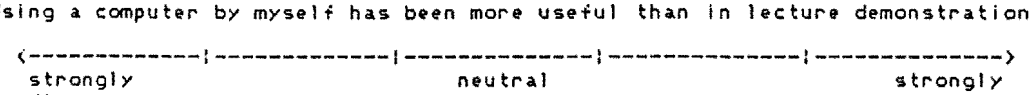
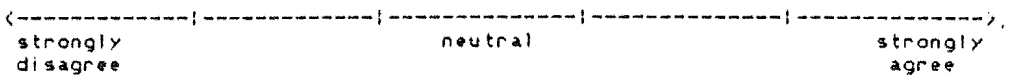
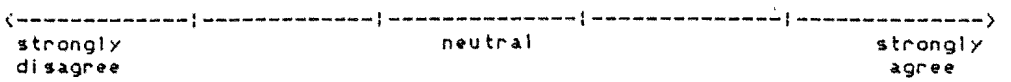


Figure 13.4a Page 1 of second attitude survey (1984). Item T and Items S1 to S5.

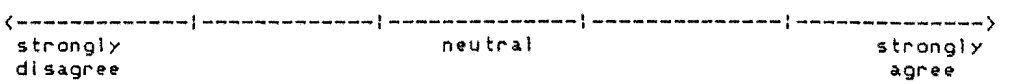
S6 Computers in education is just a gimmick.



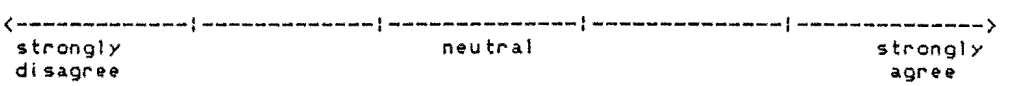
S7 Symmetry is a suitable field for computer assisted learning.



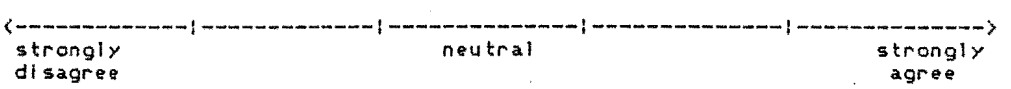
S8 I have had trouble understanding symmetry in chemistry.



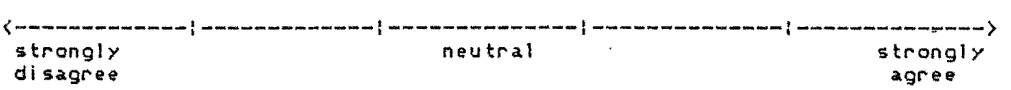
S9 I found the space group program easy to use.



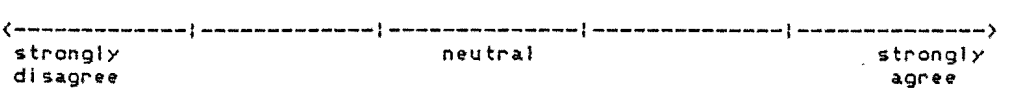
S10 The "STAF" tutorial lessons were difficult to use.



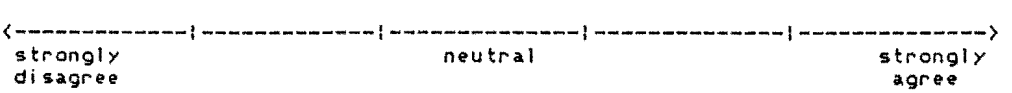
S11 I would like to use computers in future courses.



S12 Computers are not effective teaching tools in lectures.



S13 The set of assignments for the space group program was of little value.



S14 The instructions for using the space group program were adequite.

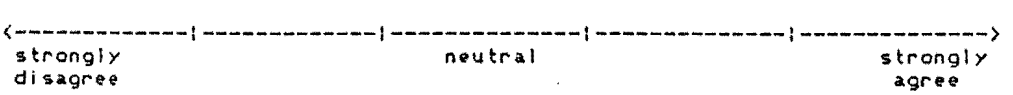
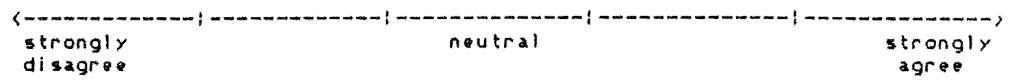
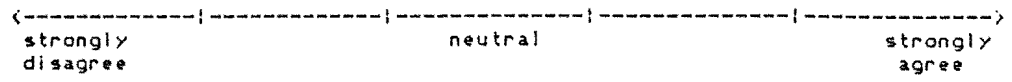


Figure 13.4b Page 2 of second attitude survey (1984). Items S6 to S14.

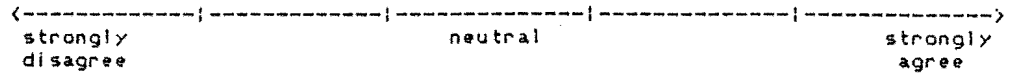
S15 The "STAF" tutorial lessons were helpful.



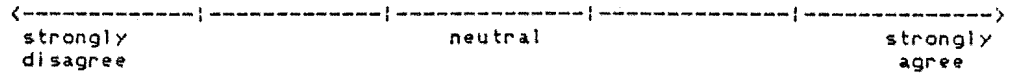
S16 It was difficult to get a computer when I wanted to use one.



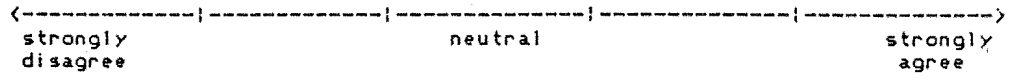
S17 I would rather work in a small group than by myself on a computer.



S18 More handouts were needed to supplement material presented on screen.



S19 Printed output of generated space groups diagrams would be desirable.



Are there any changes you would like to see in the use of computers as teaching aids (specify) ?

Record any other general comments you may wish to make about the use of computers in the teaching of the course.

Figure 13.4c Page 3 of second attitude survey (1984). Items S15 to S19.

difficulty survey are contained in table 13.1.

The variable timing of the surveys also poses problems as concept understanding appears to reduce as the delay from the end of the lecture course to the survey increases. Such a general effect is not a problem when attempting to make a qualitative identification of specific trends within a set of results but analysis to determine the statistical significance of such trends could not be performed meaningfully.

An examination of the ratios of the mean scores for each item from one year to another gives a more useful qualitative indication of trends than simple comparison of raw scores (see table 13.2). In the ratio values for the 1983 and 1981 surveys 35 of the 43 ratios have a value greater than one, indicating that those items are less well understood by the 1983 class than by the 1981 class. The ratios do however indicate some evidence of the positive effects of the computer programs used by the 1983 class. In particular the ratio values for the group of concepts relating to space groups (items C21 to C29) are noticeably lower than those for items which were not associated with any computer based teaching.

Although the concepts relating to space groups which were taught using program SGROUP provide the best support for the benefit of CAL, the group of concepts associated with point groups and individual symmetry operations (items C10 to C20) also provides favourable evidence as the use of program PGROUP to teach this parts of this area also appears

Item	Year 1981 (n=12) Mean	S.D.	1983 (n=15) Mean	S.D.	1984 (n=13) Mean	S.D.
C1	1.500	0.646	2.000	0.894	1.539	0.746
C2	2.000	0.913	2.000	0.894	1.539	0.746
C3	1.583	0.759	2.467	1.408	1.846	0.864
C4	1.667	1.027	3.000	1.095	2.231	0.799
C5	2.000	0.913	2.533	1.087	2.231	0.973
C6	2.500	0.866	3.067	0.998	3.077	1.328
C7	2.167	1.143	3.133	1.204	2.462	0.843
C8	1.417	0.640	1.600	0.611	1.385	0.625
C9	2.000	0.913	2.000	0.966	1.539	0.746
C10	1.583	0.862	1.733	0.680	1.615	0.923
C11	2.167	0.799	2.467	0.884	2.539	1.082
C12	2.000	0.817	2.133	0.957	2.385	1.595
C13	1.667	0.850	2.333	0.789	2.000	0.961
C14	2.000	0.817	2.867	1.258	-	-
C15	2.333	1.179	2.400	1.200	2.154	1.026
C16	1.833	0.986	2.533	1.147	1.923	0.829
C17	2.333	0.943	2.267	0.854	2.692	1.066
C18	2.917	0.862	2.467	0.718	2.679	1.120
C19	3.417	1.115	3.200	1.222	2.846	0.864
C20	2.417	1.320	2.667	1.075	2.539	1.216
C21	2.500	1.190	3.267	1.063	2.385	0.836
C22	2.667	1.106	3.000	0.817	2.615	1.211
C23	2.500	1.041	2.867	0.957	2.231	0.799
C24	2.750	1.422	2.733	1.063	2.308	1.202
C25	2.750	1.422	2.933	0.998	3.000	1.177
C26	3.083	1.256	2.333	0.943	2.462	0.746
C27	3.000	1.354	2.867	1.204	1.846	0.769
C28	3.083	1.038	2.733	1.289	2.231	0.697
C29	3.150	0.924	3.333	1.075	2.462	1.009
C30	1.833	0.799	2.400	1.200	1.769	0.697
C31	1.667	0.850	2.067	0.854	1.615	0.738
C32	2.417	1.038	2.800	1.108	2.154	0.864
C33	1.750	0.829	2.400	1.200	2.077	0.730
C34	1.333	0.624	2.333	1.135	1.385	0.836
C35	1.750	0.829	2.400	1.413	1.692	0.821
C36	2.083	0.759	2.467	0.957	-	-
C37	2.167	0.898	2.600	0.952	-	-
C38	2.417	1.115	2.867	1.204	-	-
C39	2.750	0.924	3.267	1.389	-	-
C40	2.500	1.118	3.200	1.166	2.231	0.973
C41	2.583	1.320	3.133	1.204	2.077	0.730
C42	2.417	0.954	3.200	1.222	2.539	0.93
C43	2.250	1.164	3.000	1.317	2.231	0.799

Table 13.1 Means and standard deviations for results of the concept difficulty survey. Items were scored on a scale of 1 to 5, with the higher number representing the greater degree of difficulty.

Item	Ratios 1983/1981	1984/1981	1984/1983
C1	1.33	1.03	0.77
C2	1.00	0.77	0.77
C3	1.56	1.17	0.75
C4	1.80	1.34	0.74
C5	1.27	1.16	0.88
C6	1.23	1.23	1.00
C7	1.41	1.13	0.80
C8	1.13	0.97	0.86
C9	1.00	0.77	0.77
C10	1.09	1.03	0.94
C11	1.14	1.17	1.03
C12	1.07	1.19	1.12
C13	1.40	1.20	0.86
C14	1.44	-	-
C15	1.03	0.92	0.90
C16	1.38	1.05	0.76
C17	0.97	1.15	1.19
C18	1.29	1.44	1.12
C19	0.97	0.83	0.86
C20	1.10	1.05	0.96
C21	1.44	0.95	0.66
C22	1.12	0.98	0.87
C23	1.15	0.89	0.78
C24	0.99	0.84	0.85
C25	1.07	1.09	1.02
C26	0.76	0.80	1.06
C27	0.96	0.62	0.64
C28	0.89	0.72	0.82
C29	1.02	0.76	0.74
C30	1.31	0.97	0.74
C31	1.28	0.97	0.76
C32	1.16	0.89	0.77
C33	1.60	1.19	0.87
C34	1.75	1.04	0.59
C35	1.37	0.97	0.70
C36	1.19	-	-
C37	1.20	-	-
C38	1.19	-	-
C39	1.19	-	-
C40	1.28	0.89	0.70
C41	1.20	0.81	0.67
C42	1.32	1.05	0.79
C43	1.33	0.99	0.74

Table 13.2 Ratios of mean results for concept difficulty survey items.

to have resulted in a reduction of ratio values relative to the areas not covered by computers. The trend for point groups would not be expected to be as great as for space groups as program SGROUP was used for the greater period of time. An isolated result in favour of the use of CAL also occurs with item C2 as this item dealing with the generation of lattices from arrays of points was dealt with using program LATTYPES.

Since the time interval from course to survey for the 1984 survey was greater than it was in 1981 but less than 1983 it would be expected that the mean values of the responses would lie between the values for the earlier surveys if no other factors were significant. Apart from the greater use of computers for teaching during the 1984 course compared to the 1983 course, some change in emphasis was present on items covered during the 1984 course. However some items were substantially unchanged, mainly in the areas of lattices and point groups not covered by computer demonstration programs. Examination of these items (C1, C3, C4 to C9, C12 and C16) does indicate that some lessening of comprehension had taken place by the time the survey was given but not to the same extent as in the case of the 1983 survey.

No major trends from 1983 to 1984 in the area of point groups and individual symmetry operations (C10 to C20) are evident as is to be expected since no changes in course content were made. The general increase in expressed comprehension in the field of space groups (C21 to C29) is

also to be expected given the increased usage of program SGROUP and its attendant revision lessons. Use of programs MILLER and BRAGG may account for the greater understanding shown for for concepts relating to diffraction of X-rays by crystals (C30 to C38) with emphasis on items C31 to C34 since these concepts are most closely associated with the programs. However the exact causes for the increase in comprehension in this subject area are difficult to pinpoint as the greater use of computers was coupled with a general stepping up of the emphasis placed on the area. A similar trend of increased understanding in a subject area may also be observed in the section dealing with structure determination (C40 to C43) but in this case it can be explained by increased coverage of the topic in the practical laboratory course to which program PATTERSON contributed in a minor role.

13.3 ATTITUDE SURVEY RESULTS

The results of the 1983 attitude survey give a clear indication that the students considered that the use of computers to teach symmetry was effective (see table 13.3). While every item in the survey relating to a specific program (items A1 to A8) indicates approval of the use of computers, the items which obtain the highest rating are those associated with the individual use programs dealing with the topic of space groups (items A6 to A8).

Responses to the preliminary attitude survey (1984)

Item	Mean	Standard Deviation
A1	3.923	0.954
A2	3.933	0.961
A3	3.500	0.760
A4	3.750	0.965
A5	3.889	1.167
A6	4.214	0.802
A7	4.364	0.674
A8	4.100	0.568
A9	4.143	0.864
A10	3.286	1.204
A11	3.929	0.616
A12	3.857	1.027
A13	3.385	0.650

Table 13.3 Results of the 1983 attitude survey. Items were scored on a scale of 1 to 5 with 1 representing the most negative attitude and 5 the most positive.

Item	Score Order	Mean	Standard Deviation
P1	+	3.588	0.772
P2	-	3.588	0.974
P3	+	4.000	0.686
P4	-	2.765	0.807
P5	-	2.412	1.240
P6	-	3.882	1.022
P7	+	3.647	1.185
P8	-	4.294	0.892
P9	+	2.824	1.339

Table 13.4 Results of the preliminary (1984) attitude survey. Items were scored on a scale of 1 to 5. Items with score order "+" were scored with strong agreement as 5 while items with score order "-" were scored with strong disagreement as 5.

tended to be neutral in nature with few strong attitudes expressed (see table 13.4). The two most notable results to come from this survey were clear indications that the students felt that symmetry is a suitable field for computer assisted learning (item P8) and that computers would enhance students ability to comprehend symmetry (item P3).

The second attitude survey tended to indicate stronger attitudes than those expressed in the first survey presumably as a result of the students recent experience with the matters being surveyed (see table 13.5). The results with the strongest attitudes, as indicated by the largest scores, were those which dealt directly with the usefulness of CAL, which show that the students found using computers had a significant effect in the learning of symmetry concepts in chemistry. Results of particular note which show strong attitudes in favour of the use of computers are those which correspond to: the use of computers in education not being a gimmick (item S6), that symmetry is a suitable field for CAL (S7), that computers provide enhanced comprehension of crystal symmetry (S3) and that computer use in future courses would be desirable (S11).

Examination of the results of items dealing with specific details of the computer content in the symmetry course indicates that students have a definite preference for using computers on an individual basis rather than in lecture demonstrations (item S5) although the clear indication that computers are an effective teaching tool in

Item	Score Order	Mean	Standard Deviation
S1	+	4.154	0.864
S2	-	3.462	1.009
S3	+	4.077	0.829
S4	+	3.000	1.240
S5	-	1.846	1.099
S6	-	4.769	0.421
S7	+	4.462	0.634
S8	-	3.077	1.141
S9	+	3.692	1.202
S10	-	3.385	0.836
S11	+	4.077	1.071
S12	-	4.000	0.877
S13	-	4.154	0.662
S14	+	4.000	0.877
S15	+	4.000	0.877
S16	-	4.077	0.829
S17	-	3.231	1.367
S18	+	3.769	1.120
S19	+	3.923	0.829
T (Total time spent using computers)		4.077	1.639

Table 13.5 Results of the second (1984) attitude survey. Items were scored as for the preliminary survey (Table 13.4).

lectures (S12) should also be noted when considering this result. The assignments for SGROUP (S13) and the tutorial lessons (S15) were found to be useful although the ease of use ratings for them (S9 and S10) were only slightly better than neutral indicating a need for more user friendly programming or better instructions. The rating for the SGROUP instructions did however indicate a reasonable level of approval (S14).

A series of correlations were made between selected items from the concept difficulty survey and items from the second attitude survey for the 1984 class [64]. The items from the attitude survey are the six most directly related to the use of computers (items S1, S3, S4, S9, S13 and S15) together with the total hours spent using the computer (item T). The items from the concept difficulty survey fall into two areas, the first of which contains five items which had shown a significant rise in the level of understanding from the previous years (items C9, C16, C21, C27 and C32). These items all directly related to material which had been covered using computers. The second area consists of items that had shown little change from previous years or had little connection with the material presented by computer (items C6, C11, C22 to C25, C35 and C42).

While the small sample size used to calculate the correlations given in table 13.6 means that any specific correlation value may be spurious, it is possible to draw inferences from trends present in groups of correlations. To illustrate this, examination of the correlations between

	T	S1	S3	S4	S9	S13	S15	C9	C16	C21
S1	*									
S3	*	0.629								
S4	-0.452	*	*							
S9	*	*	*	0.332						
S13	*	*	*	*	*					
S15	0.535	*	*	*	*	0.398				
C9	*	*	-0.316	*	-0.588	-0.324	-0.470			
C16	*	*	-0.440	-0.540	-0.565	*	*	*		
C21	*	*	*	*	*	-0.524	-0.315	0.532	*	
C27	*	-0.312	-0.585	-0.350	-0.634	*	*	0.413	0.706	0.331
C32	*	-0.341	-0.662	-0.340	-0.548	*	*	0.469	0.662	*
C6	*	0.325	*	-0.463	-0.371	*	*	0.580	0.425	0.389
C11	0.670	*	*	-0.457	-0.464	0.636	0.405	*	0.561	*
C22	*	*	-0.507	*	*	*	0.362	*	0.430	0.450
C23	*	*	*	-0.668	-0.647	*	-0.329	0.308	0.724	*
C24	*	*	*	-0.488	-0.734	*	*	*	0.796	*
C25	-0.319	*	*	0.319	-0.326	*	*	*	*	*
C35	*	-0.693	-0.530	0.334	*	-0.338	*	*	*	*
C42	*	*	-0.553	*	*	*	*	*	0.653	*

	C27	C32	C6	C11	C22	C23	C24	C25	C35
C32	0.846								
C6	*	*							
C11	*	*	*						
C22	0.432	0.351	0.353	*					
C23	0.558	0.506	0.346	0.390	*				
C24	0.551	0.548	0.515	0.405	0.451	0.807			
C25	*	*	*	*	0.486	*	0.544		
C35	*	*	-0.331	-0.333	0.345	-0.360	*	0.318	
C42	0.654	0.568	*	0.323	0.799	*	0.403	*	0.318

Table 13.6 Correlations between selected survey items. Correlations with an absolute value less than 0.3 were suppressed as indicated by the presence of the "*" symbols.

items from the concept difficulty survey show a high degree of correlation as would be expected from a set of items so closely related. When examining values for correlations between items from the concept difficulty survey and items from the attitude survey it should be noted that because of the scoring system used a negative value for the correlation corresponds to a positive relationship between the strength of the attitude and how well the concept is understood.

Inspection of the table of correlations reveals that while a proportionately greater number of the more significant positive correlations appear between attitude survey items and the first group of concept difficulty items there are also significant correlations present with the second group. The correlations between the ease of use of SGROUP (item S9) and those items associated with space groups (C22, C23 and C24) are however not unexpected given the related nature of the items. This leads to the conclusion either that knowledge of space groups makes SGROUP an easier program to use or that use of the program aids comprehension of the subject area but not to the extent that changes in overall class understanding were readily discernible. More obvious trends are observable with items C27 and C32 which do show a noticeable increase in comprehension and exhibit clear correlations with attitude items related to computer use (S1, S3, S4 and S9).

13.4 SURVEY CONCLUSIONS

The qualitative analysis of all results from the surveys undertaken does provide clear evidence in favour of the continued use of computer based teaching methods in the area of crystal symmetry and related fields. The concept difficulty survey was "results based" in that it dealt with how well the subject was understood at the completion of the course so the results from these surveys in effect measure, in part, the effectiveness of the teaching methods. Although timing factors prevented quantitative statistical analysis, inspection of the results clearly showed a relative increase in understanding in those areas where computers had been introduced into the course thereby indicating a greater effectiveness through computer based teaching methods than through the traditional methods used previously.

The attitude survey results indicated that students were favourably disposed towards the use of computers to teach chemistry. The survey results indicate that students found all aspects of the computer sections of the course helpful with the individual use sections being of the greatest benefit. Although some of the enthusiasm shown by the students may be accounted for by the novelty of using computers it is significant that the attitudes expressed in the second attitude survey (prefix S) were more generally positive in nature than those expressed in the earlier survey (prefix P) indicating that at least part of the

enthusiasm expressed is because of the use of computers during the course. Of particular note is the clear indication that students would like to use computers in future courses (item S11). The implied motivational aspect of this result is the best single indication that computer based teaching has a part to play in chemical education in the future.

CHAPTER XIV

CONCLUSIONS

This project was designed to find ways in which computer assisted learning techniques could be applied to a particular area of chemistry to greatest advantage. Achievement of the desired objectives necessitated the production of a substantial ammount of software and assessment of its effectiveness as measured by student response. In attempting to draw final conclusions we have placed emphasis more on the merits of the various types of teaching program used than on the performance of specific programs so that our findings may be of general use when planning future CAL software development.

14.1 OVERALL EFFECT OF CAL PROGRAMS

The use of computer graphics for lecture demonstration purposes fulfilled most of our early expectations for it. Programs were written which could demonstrate aspects of symmetry and crystallography in a lecture in ways not possible using conventional techniques. Specifically these programs were able to generate, in any order, a large number of possible graphics displays in a stepwise fashion, while using animation, colour and other special effects for emphasis or to better illustrate a point. Teaching plans based on these programs could be constructed and attitude

surveys of students exposed to computer based teaching in lectures indicated that they found such teaching methods worthwhile. It may therefore be concluded that computer based lecture demonstration programs can be successfully written and that they can be used to advantage in the teaching of various aspects of chemistry.

Similar conclusions can be drawn from the subsequent use of demonstration programs in an auto-elaborative role. With the provision of suitable guiding material this has proved an effective way of allowing students to explore a subject rather than having the subject taught to them. Attitude survey results, taken after students had been allowed free use of these programs, confirm the usefulness of this technique. The results showed this style of computer based teaching to be the most popular.

The drill and practice and tutorial programs using more established teaching techniques produced little new information on their own. From their use in conjunction with other programs however it was shown to be possible to construct teaching packages to cover all phases of the teaching of a particular subject area. The fact that such an integrated software package is an effective teaching tool is a significant result since it emphasises the value of a comprehensive coverage of an aspect of chemistry through the use of CAL techniques as compared with more common piecemeal approaches.

14.2 THE TEACHING SYSTEMS USED

The development of any CAL material must always be influenced by the hardware and software systems used to implement it. Although careful consideration of the choice of systems must necessarily be made prior to the commencement of program development, a final evaluation can only be given after programming has been completed.

The Apple // series of computers, although now technologically outdated, proved to be very suitable for CAL purposes mainly as a result of the large amounts of general software, especially languages, which have been developed for it and the way it can be adapted to many different roles. The graphics capabilities of the Apple are adequate for most purposes and it is only the lack of sufficient memory and processing speed for larger and more complex applications which will count against it for future applications.

The development of programs during this project was greatly aided by the use of a structured programming language. Thus, Pascal proved ideal for large programs since it leads naturally to the development of a logical, organised and readable program code. Implementations of Pascal or other structured languages such as C or Modula 2 are now available on all but the smallest computer systems and it is to be hoped that they will be used in future CAL developments in situations where a general purpose programming language is to be used.

Of the authoring languages examined, none proved worthy of recommendation for future use in their current forms. Although the MASTER system was used and proved satisfactory in the limited role required of it, it is not sufficiently generalized to be used in most tutorial applications. STAF has many very powerful features but its use is limited by the lack of suitable graphics facilities. Likewise PILOT as a language has much to recommend it but the implementations on the Apple are such that its features cannot be used to their full potential.

14.3 THE FUTURE OF CAL IN CHEMISTRY

As more powerful computer hardware becomes readily available the limits of usefulness of CAL are going to be increasingly dependent on educators' willingness to develop and use software which exploits the capacities of this hardware. While current state of the art hardware technology is adequate for most foreseeable software developments, effort is still necessary to make some newer products easier to use. Also cost still limits the use of some features, particularly those related to memory and peripheral storage.

Laser disk technology currently used in arcade video games may be used to enhance the graphics capabilities of small computers. However in most applications in chemistry it is probable that a standard computer graphics display with resolution comparable to current specialized graphics

terminals will meet any teaching requirements. This is because most chemical concepts may be presented in a diagrammatic manner and therefore may be handled by relatively simple graphics systems.

One of the problems associated with greater graphics capabilities is that with higher resolution the displays are of greater potential complexity and require increased programming time to develop. This problem can be overcome only through the availability of suitable utilities written to facilitate more simple development of lessons. This highlights another problem which effectively bars any widespread introduction of CAL into chemistry in the short term. The problem is that chemistry teachers tend not to be proficient in programming of the type required for CAL while the majority of specialist computer programmers are not knowledgeable in both chemistry and education skills. In the long term this may become less of a problem because of the expected general rise in computer literacy as a result of computers becoming widespread throughout society, but in the meantime development must be restricted to those chemists who have the necessary skills.

An important probable consequence of this limit in potential CAL output is that the use of demonstration, simulation and auto-elaborative programs should increase relative to that of tutorial and drill and practice programs. The former group is made up of programs which are more flexible in nature and should therefore be able to be used by a larger number of teachers than the programs in the

second group. These other programs are usually written for a specific course and so may not translate well to another teacher's method of teaching a subject.

A converse argument is that while authoring systems for tutorial and drill and practice programs are currently rather primitive and therefore require computer programming skills to use them, this may not always be the case and it should be possible in the near future for an authoring system to be developed which requires little or no computing skill. Such a system would ideally require no knowledge of any operating system but rather would be a program which would prompt the teacher for the material needed to construct a lesson, such as explanatory text, questions, possible responses, answers, and lesson flow information. From this information the program should be able to create and package a lesson ready to be tested and used.

While utilities exist to aid in the writing of some types of teaching programs, it is unlikely in the short term that such a "hand holding" system can be developed for any programs incorporating dynamic graphics. The number of variations for a program of any complexity would be too large for any generalized development program. However this does not mean that development programs cannot be written which would at least reduce the dependence on programming skills by the teacher. These development programs may be considered as user friendly extensions of current programming and graphics utility programs. When sufficiently flexible development utilities can be developed

then the production of demonstration and auto-elaborative programs will be greatly facilitated. Once such programs are readily available, their inherently adaptable nature means they should play a large and significant role in the future expansion of CAL in the field of chemical education.

APPENDIX A

PACKAGING OF PROGRAMS

If the best use is to be made of any software it is important that it is packaged in such a way that is convenient for the intended user. A program should be easily executable, and require only the minimum knowledge of the computer operating system. Otherwise it may not be used by the full range of people for whom it is suitable. This is of particular concern for educational software where the programs are to be used by teachers and students who may have little or no computer experience.

A.1 REQUIREMENTS OF PROGRAMS.

All the programs written in this project, run under the Apple-UCSD-Pascal 1.1 operating system. This requires that a number of Pascal system files are present when the programs are executed. If a program is to run using the minimum hardware configuration, specified in chapter 5, then the system files, the program and any data files must all reside on one 5 1/4 inch floppy disk.

Each disk may contain 140k bytes storage, divided into 280 blocks. The directory of the disk occupies the first six blocks, and a further 76 blocks are occupied by the Pascal system files necessary for all applications. In addition the system library, which contains turtlegraphics

and other utilities, may occupy up to 34 blocks. By deleting those parts not used by any program the library can be reduced to 25 blocks. This means that a maximum of 173 blocks are available for programs and data files.

In practice short programs to provide greeting messages, enable lower case chips, or provide menus reduce this number by two to six blocks. The MASTER interpreter, as described in chapter 6, must be present on any disk running a MASTER lesson, thereby accounting for another fourteen blocks. If the system has a second disk drive, a significantly larger amount of disk space is available as a disk in the second drive need not contain any system files which are provided on the "boot disk" contained in the first drive.

The Apple][and Apple][+ model computers were not able to display lower case letters without a hardware modification. The Apple //e however has a lower case capability as a standard feature and lower case lettering has been used whenever appropriate to make larger blocks of text easier to read. If a program containing lower case text is executed on an Apple][or Apple][+ that does not have the lower case modification the lower case characters will be displayed as upper case characters. When such a program is executed on a system with the lower case modification the lower case characters will display as upper case unless a short software patch is executed, after which the text will be displayed correctly [29]. If this patch is used on a machine without a lower case facility the output

is rendered meaningless. To overcome this difficulty a short procedure was written which inquires of the user if a lower case chip is present. A positive response causes the appropriate software patch to be activated.

Expansion cards to generate an 80 column text display are a common option on Apple two series computers. Unfortunately several models of 80 column card do not allow the display of graphics, while those used on the Apple //e interfere with the operation of two page animated graphics techniques, such as those used by program SYMPACK (see chapter 10). A solution to this problem has been to modify one of the system files (SYSTEM.APPLE) so that the 80 column card is not switched on.

A.2 PACKAGING OF PROGRAM SGROUP

Program SGROUP (see chapter 7) was the first program to be packaged for distribution to users both outside and within the University of Canterbury. The size of the code file for SGROUP has never exceeded 45 blocks of disk space, hence packaging an executable version of the program alone caused no space problems. However the distribution package was to include source files and the system text editor and file handling system as well as the minimum set of system files. Since information may be stored on both sides of the floppy disks used on the Apple it was decided to place all the system files and a copy of the code file on one side, with the source files together with another copy of the code

file on the reverse side.

The earlier versions of SGROUP were written in Pascal version 1.0 which does not permit execution of one program from within another. This process, known as chaining, is useful for the setting up of a menu from which the user can select the desired program which is then automatically executed. In the earlier version the disk containing the system files was set up to boot in the standard manner to the Apple-pascal command mode. The user was supplied with instruction as to how to execute the program. After conversion to Pascal system 1.1 an optional lower case patch was added to the package, but the program execution was left unchanged to conform to the already existing documentation.

When the symmetry operations package was assembled, SGROUP was included so that the disk containing system files in the original release no longer necessary. The disk containing source files was retained for distribution only when requested, the reverse side of the symmetry operations disk being reserved for utility programs and documentation.

A.3 THE SYMMETRY OPERATIONS PACKAGE

The symmetry operations package consists of nine lecture demonstration programs, collected onto a single disk. The purpose of packaging the programs in this way was to make them easier to use in combination, and to simplify handling and distribution. A list of the nine teaching programs is given in table A.1.

Program Name	Filename on disk	Chapter Reference
Point Group	PGROUP	10
Space Group	SGROUP	7
Patterson Plot	PATTERSON	9
Symmetry Operations	SYMPACK	10
Bragg Reflection	BRAGG	11
Miller Plane	MILLER	11
Plane Patterns	SEVENTEEN	11
Bravais Lattices	LATTYPES	11
Unit Cells	TWOD	11

Table A.1 Program and file names for teaching programs contained in the Symmetry Operations Package.

When assembling the earlier versions of the package it was possible to fit the code files of all nine programs on one side of a 5 1/4 floppy disk by removing all non-essential files and reducing the system library file to the minimum possible size. All documentation for these early versions was included as separate hard copy with no disk files containing instructions.

As later versions of the constituent programs contained features not present in their predecessors, the code files for the programs became larger. More documentation for the package was also written and saved in disk files. Finally two new utility programs for program SGROUP were written. These programs were PACKER and MOLEDIT which are described in chapter 7.

To accommodate the need for extra disk space the package was altered so that both sides of the floppy disk were used. The original symmetry operations package, minus the Bravais lattices program (LATTYPES), was placed on one side of the package, and the utilities, documentation and the Bravais lattices program placed on the second side. The two sides of the disk both contain a minimum set of system files to boot the disk and run the programs, so ensuring that the package will continue to be executable with the minimum hardware configuration. Directory listings of both sides are contained in figures A.1 and A.2.

Side one of the symmetry operations package contains program SYMPACK which uses two page animated graphics techniques, hence that side of the disk must use the

```
KODE:
SYSTEM.APPLE        32 21-Jun-83     6    512  Datafile
SYSTEM.LIBRARY      25 14-Aug-80    38    512  Datafile
SYSTEM.PASCAL       41 10-Dec-82    63    512  Datafile
SYSTEM.CHARSET       2  7-Oct-83   104    512  Datafile
SYSTEM.MISCINFO      1 19-Sep-82   106    192  Datafile
PTGP.DATA            2 29-Jul-81   107     28  Datafile
CELL.DATA            1  4-Aug-81   109     52  Datafile
QMK.DATA             1  5-Nov-81   110     96  Datafile
CSHPS.DATA           1 12-Jan-83   111    476  Datafile
SYSTEM.STARTUP       2 28-Jun-83   112    512  Codefile
PGROUP.CODE         20 20-Jul-84   114    512  Codefile
PKSHPS.DATA          1 28-Jun-83   134     44  Datafile
MILLER.CODE          5 20-Jul-84   135    512  Codefile
SGROUP.CODE         45 20-Jul-84   140    512  Codefile
CUACAC.DATA          1 13-Jul-83   185    237  Datafile
SYMPACK.CODE        21 30-Jul-84   186    512  Codefile
SHAPES.DATA          1 12-Jan-83   207    220  Datafile
BENZAMIDE.DATA       1 17-Jan-84   208    186  Datafile
SEVENTEEN.CODE       8 30-Jul-84   209    512  Codefile
MENU.CODE            3 18-Jun-84   217    512  Codefile
PATTERSON.CODE      40 20-Jun-84   220    512  Codefile
BRAGG.CODE           5 20-Jul-84   260    512  Codefile
TWOD.CODE            9 30-Jul-84   265    512  Codefile
< UNUSED >           6             274
23/23 files, 274 blocks used, 6 unused, 6 in largest
```

Figure A.1 Directory listing of side 1 of the Symmetry Operations disk.

```
DOC:
SYSTEM.LIBRARY     34 19-Sep-80     6    512  Datafile
SYSTEM.APPLE       32 18-Nov-82    40    512  Datafile
SYSTEM.PASCAL      41 10-Dec-82    72    512  Datafile
SYSTEM.MISCINFO     1 19-Sep-82   113    192  Datafile
SYSTEM.STARTUP      2 16-Jan-84   114    512  Codefile
PACKER.TEXT         6 17-Jan-84   116    512  Textfile
PACKER.CODE         3 17-Jan-84   122    512  Codefile
LISTER.TEXT         8 17-Jan-84   125    512  Textfile
PATINFO.TEXT       26 29-Jun-84   133    512  Textfile
LATTYPES.CODE      11 31-Jul-84   159    512  Codefile
MOLEDIT.CODE       11 31-Jul-84   170    512  Codefile
SYSTEM.CHARSET      2  7-Oct-83   181    512  Datafile
MOLEDIT.TEXT       16 31-Jul-84   183    512  Textfile
SGINFO.TEXT        26 30-Jul-84   199    512  Textfile
CUACAC.DATA         1 13-Jul-83   225    237  Datafile
LISTER.CODE         4 17-Jan-84   226    512  Codefile
BENZAMIDE.DATA      1 17-Jan-84   230    186  Datafile
EDINFO.TEXT        10 19-Jul-84   231    512  Textfile
SYMINFO.TEXT       14 14-Sep-84   241    512  Textfile
PGINFO.TEXT        16 30-Jul-84   255    512  Textfile
MENU.TEXT           4 18-Jun-84   271    512  Textfile
MENU.CODE           3 18-Jun-84   275    512  Codefile
< UNUSED >          2             278
22/22 files, 278 blocks used, 2 unused, 2 in largest
```

Figure A.2 Directory listing of side 2 of the Symmetry operations disk.

modified SYSTEM.APPLE file to deactivate the 80 column card. However the modified file is not used on side two of the disk as the documentation contained on the disk is configured for an 80 column display if one is installed.

A.4 PRESENTATION AND DOCUMENTATION

When either side of the symmetry operations package is booted the user is first asked a question relating to a lower case chip, as described in section A.1. Following this a menu of the available programs is presented. An example of such a menu display is given in figure A.3.

The major advantage gained by using a menu program is that the user is isolated from the operating system. The user therefore need not be familiar with the operating system or any of its command sequences. The only knowledge required is the information that pressing the appropriate key causes the desired program to be executed.

The filename corresponding to a particular program may not be meaningful to the user since the length of the name and the characters used in the name are both restricted by the system. The Apple-Pascal system effectively allows only ten characters in a name with no spaces or underline characters. Use of a menu program means the program user need not know the program name, as the desired program may be chosen from the description in the menu.

A single page of instructions explains the startup procedure, and describes how further program documentation

SYMMETRY OPERATIONS
-------- ----------

The programs are

1 : POINT GROUP
2 : SPACE GROUP
3 : PATTERSON PLOT
4 : BRAGG REFLECTION
5 : MILLER PLANE
6 : SYMMETRY OPERATIONS
7 : PLANE PATTERNS
8 : UNIT CELLS

0 : Return to command mode

Input the number of your choice

Figure A.3 Menu display generated by the startup program of the Symmetry Operations disk (side 1).

may be obtained. All other documentation is contained on side two of the disk. A utility program (LISTER) is provided to list any of the five files making up the documentation. The files may be listed to either the screen or to a printer.

A.5 OTHER PACKAGES

Two special purpose packages designed for use within the University of Canterbury have also been constructed. The first of these is a version of the symmetry operations package containing only four programs on a single disk side. The four programs are PGROUP, SGROUP, PATTERSON and SYMPACK. The other programs usually found in the symmetry operations package were removed to provide space for data files containing molecular information for programs SGROUP and PATTERSON. This package was intended for use by students in the honours part II class, to be used to complete exercises involving the added data files used with SGROUP.

The second package was made up of four tutorial programs, three of which were written in the MASTER teaching language and the other in Pascal. Two of these were followup lessons to program SGROUP, while the other two were lessons for use in conjunction with program SYMPACK. The content of these lessons is covered in chapters 8 and 10.

Since tutorial lessons written in the MASTER system require a considerable amount of disk storage it was not possible to create a single disk system for the tutorial

package. To illustrate this consider the example of the tutorial relating to space group diagrams. A full size system library is required to run the master system so only 164 blocks of disk space were available once space for system files was allocated. A further 25 blocks was required for the MASTER interpreter and the greeting and menu programs. The lesson in its most compact form requires 79 blocks of space with a further 8 blocks required to be left free to accommodate response files. This leaves 52 blocks free for the other three programs, any one of which would require most of that space.

It was therefore decided to create a two disk package, thereby providing more than ample disk space for all files including source files not essential to the execution of the lessons. Directory listings of the disks are contained in figures A.4 and A.5. It should be noted that the MASTER interpreter (INTERP.CODE) is to be found on both disks. The purpose of this duplication is to minimize the amount of disk usage in the interests of rapid program execution.

```
STAF:
SYSTEM.APPLE        32 21-Jun-83     6    512  Datafile
SYSTEM.LIBRARY      34 19-Sep-80    38    512  Datafile
SYSTEM.PASCAL       41 10-Dec-82    72    512  Datafile
RESTART.CODE         3  4-Jul-84   113    512  Codefile
SYSTEM.CHARSET       2  7-Oct-83   116    512  Datafile
LOGO.FSCR            6 28-Feb-83   118    512  Datafile
SYSTEM.MISCINFO      1 16-Aug-80   124    192  Datafile
PMNA!.FSCR           4 29-Nov-82   125    512  Datafile
P21!C.FSCR           4 29-Nov-82   129    512  Datafile
PBCN!.FSCR           4 29-Nov-82   133    512  Datafile
PBCN3.FSCR           3  1-Dec-82   137    512  Datafile
PBCN2.FSCR           4  1-Dec-82   140    512  Datafile
INTERP.CODE         14 14-Dec-82   144    512  Codefile
SYSTEM.STARTUP       2 25-Jul-83   158    512  Codefile
SGSTAF.OP            2 27-Aug-84   160    176  Datafile
SGSTAF.NODE          2 27-Aug-84   162    218  Datafile
PMNA.TEXT           10 27-Aug-84   164    512  Textfile
PMNA.STR             8 27-Aug-84   174    490  Datafile
PMNA.OP              1 27-Aug-84   182    356  Datafile
PMNA.NODE            1 27-Aug-84   183    380  Datafile
PMNA2.TEXT          14 27-Aug-84   184    512  Textfile
PMNA2.STR           11 27-Aug-84   198     46  Datafile
PMNA2.OP             1 27-Aug-84   209    444  Datafile
PMNA2.NODE           1 27-Aug-84   210    500  Datafile
PBCN.STR            13 10-Aug-84   211    114  Datafile
PBCN.OP              1 10-Aug-84   224    484  Datafile
PBCN.NODE            2 10-Aug-84   225     38  Datafile
SGSTAF.STR          17 27-Aug-84   227    250  Datafile
PBCN.STAT            2 14-Aug-80   244    454  Datafile
SGSTAF.TEXT         20 27-Aug-84   246    512  Textfile
< UNUSED >          14             266
30/30 files, 266 blocks used, 14 unused, 14 in largest
```

Figure A.4 Directory listing of the STAF: disk. This disk is the one with which the tutorial system is booted.

```
STAF2:
RECOVER.CODE          3  1-Jun-82      6    512  Codefile
REF.FSCR              2  2-Dec-82      9    512  Datafile
INV.FSCR              2  2-Dec-82     11    512  Datafile
TRI.TEXT              4  6-Dec-82     13    512  Textfile
REF2.FSCR             3  8-Dec-82     17    512  Datafile
GLI.FSCR              2  8-Dec-82     20    512  Datafile
QMARK.TEXT           10  8-Dec-82     22    512  Textfile
REF3.FSCR             3  8-Dec-82     32    512  Datafile
ROT.FSCR              1  9-Dec-82     35    512  Datafile
SCR.FSCR              2 13-Dec-82     36    512  Datafile
TRI2.TEXT            10 13-Dec-82     38    512  Textfile
TEACHER.CODE         16 10-Dec-82     48    512  Codefile
PIC1.FSCR             3 20-Jan-83     64    512  Datafile
PIC2.FSCR             2 24-Jan-83     67    512  Datafile
PIC3.FSCR             2 24-Jan-83     69    512  Datafile
STAFANAL.CODE        17 27-Jan-83     71    512  Codefile
PIC4.FSCR             2 31-Jan-83     88    512  Datafile
PIC5.FSCR             2  1-Feb-83     90    512  Datafile
STAFINST.TEXT         4  1-Feb-83     92    512  Textfile
CIRC2.FSCR            1  3-Feb-83     96    512  Datafile
CIRC1.FSCR            2  3-Feb-83     97    512  Datafile
QMK.DATA              1  5-Nov-81     99     96  Datafile
TFIL2.DATA            9 11-Feb-83    100    104  Datafile
TUTORIAL.CODE        25 14-Feb-83    109    512  Codefile
SYMOP.TEXT           18 11-Jul-83    134    512  Textfile
SYMOP.OP              2 11-Jul-83    152    112  Datafile
SYMOP.NODE            2 11-Jul-83    154    128  Datafile
HOWTO.TEXT            4 14-Feb-83    156    512  Textfile
SYMOP.STR            15 11-Jul-83    160     56  Datafile
TFILE.DATA            9 11-Feb-83    175    104  Datafile
CCORD.OP              1  9-Aug-84    184    512  Datafile
CCORD.NODE            2  9-Aug-84    185     28  Datafile
INTERP.CODE          14 14-Dec-82    187    512  Codefile
CCORD.STR            15  9-Aug-84    201     56  Datafile
RESTART.TEXT          4  4-Jul-84    216    512  Textfile
CCORD.TEXT           18  3-Aug-84    220    512  Textfile
PBCN.TEXT            16  9-Aug-84    238    512  Textfile
CCORD.STAT            1 14-Aug-80    254    420  Datafile
SYMOP.STAT            1 14-Aug-80    255    504  Datafile
RESP.TEXT             4 14-Aug-80    256    115  Textfile
< UNUSED >           20              260
40/40 files, 260 blocks used, 20 unused, 20 in largest
```

Figure A.5 Directory listing of the STAF2: disk. This disk is placed in the second drive when the system is booted with the STAF: disk.

REFERENCES

1 McKenzie, J., Comput. Educ., 2, 25, (1978).

2 Moore, J.W., Gerhold, G., Bishop, R.D., Gelder, J.I., Pollnow, G.F. and Owen, G.S., J. Chem. Educ., 57, 93, (1980).

3 Dence, M., Educ. Tech. Nov 1980, 50.

4 Beard, R.M., Chem. Brit., 7, 324, (1971).

5 Hooper, R., Comput. Educ., 2, 1, (1978).

6 Whiting, J., Comput. Educ., 9, 101, (1985).

7 Gagne, R.M., Wager, W. and Rojas, A., Educ. Tech., Sep 1981, 17.

8 Laurillard, D., Comput. Educ., 2, 259, (1978).

9 Boyd, G.M., Comput. Educ., 6, 305, (1982).

10 Papert, S. "Mindstorms," The Harvester Press, Brighton, Sussex, 1980.

11 Skyme, D.J., Comput. Educ., 6, 13, (1982).

12 Soltzberg, L.J., J. Chem. Educ., 49, 357, (1972).

13 Jenkin, J.M., Comput. Educ., 6, 25, (1982).

14 Ayscough, P.B., Brit. J. Ed. Tech., 8, 201, (1977).

15 Bork, A., "Learning With Computers," Digital Press, Bedford, Massachusetts, 1981.

16 McCulloch, D.W. Educ. Tech., Jun 1980, 12.

17 "Reviews," J. Chem. Educ., 60, A179, (1983).

18 Ayscough, P.B., Chem. Brit., 12, 348, (1976).

19 Cotton, F.A., "Chemical Applications of Group Theory," (second edition), Wiley-Interscience, New York, 1971.

20 Carpenter, G.B., "Principles of Crystal Structure Determination," W.A. Benjamin inc, New York, 1969.

21 Hathaway, B.J., J. Chem. Educ., 56, 166, (1979).

22 House, D.A., Hay, R.W. and Akbar Ali, M., Inorg. Chim. Acta., 72, 239, (1983).

23 Bang, E. and Monsted, O., Acta Chem. Scand. Ser. A, 36, 353, (1982).

24 Eisele, J.E., Educ. Tech., Oct 1979, 13.

25 England, E., Comput. Educ., 8, 317, (1984).

26 Moore, J.W., Gerhold, G., Breneman, G.L., Owen, G.S., Butler, W., Smith, S.G. and Lyndrup, M.L., J. Chem. Educ., 56, 776, (1979).

27 Owen, G.S., J. Chem. Educ., 61, 440, (1984).

28 "Applesoft II BASIC Programming Reference Manual," Apple Computer inc, Cupertino, California, 1979.

29 Lancaster D., Microcomputing, Nov 1979, 30.

30 Curtiss, D., Call -apple, Jun 1980, 33, A.P.P.L.E. co-op, Renton, Washington.

31 "Microsoft Softcard," Microsoft Consumer Products, Bellevue, Washington, 1980.

32 Jensen, K. and Wirth, N., "Pascal User Manual and Report," (second edition), Springer Verlag, New York, 1976.

33 Wakerly, J., Microcomputers and Microsystems, 3, 323, (1979).

34 Alpert, S.R., Byte, Aug 1978, 78.

35 "Apple Pascal Operating System Reference Manual," Apple Computer inc, Cupertino, California, 1980.

36 Mundie, D.A., Byte, Aug 1979, 110.

37 Keyser, E., Call -apple, Oct 1983, 55, A.P.P.L.E. co-op, Renton, Washington.

38 "Apple PILOT Language Reference Manual," Apple Computer inc, Cupertino, California, 1980.

39 Nelson, L.R., "18th New Zealand Electronics Conference," 1981. p.13-20.

40 Barker, P.G. and Singh, R., Brit. J. Ed. Tech., 14, 174, (1983).

41 Ayscough P.B., Chem. Brit., 9, 61, (1973).

42 Barker, P.G. and Wilford, J., Brit. J. Ed. Tech., 16, 115, (1985).

43 "Prodos Reference Manual," Apple Computer inc, Cupertino, California, 1983.

44 Robinson, A.H., Call -apple, Oct 1980, 23, A.P.P.L.E. co-op, Renton, Washington.

45 Johnson, S.C. and Kernighan, B.W., Byte, Aug 1983, 48.

46 Hockenhull, J., Call -apple, Dec 1983, 9, A.P.P.L.E. co-op, Renton, Washington.

47 Abelson, H., "Apple Logo," Byte/McGraw-Hill, Peterborough, New Hampshire, 1982.

48 Sandys, J., Call -apple, Aug 1984, 29, A.P.P.L.E. co-op, Renton, Washington.

49 Harper, J., Call apple, Aug 1984, 49, A.P.P.L.E. co-op, Renton, Washington.

50 Ross, A.W., "Microcomputer Presentation of Lessons in Chemistry," University of Canterbury, Christchurch, (Thesis MSc Chemistry).

51 Bowles, K.L., Byte, Aug 1978, 46.

52 Penfold, B.R. and Temple, R.S., J. Chem. Educ., 59, 775, (1982).

53 "International Tables for X-ray Crystallography," Vol I, Kynoch Press, Birmingham, England, 1952.

54 Buerger, M.J., "Elementary Crystallography," John Wiley and Sons inc, New York, 1956.

55 Penfold, B.R. and White, J.C.B., Acta. Cryst., 12, 130, (1959).

56 Di Blasio, B., Pavone, V. and Pedone, C., Cryst. Struct. Comm., 6, 845, (1977).

57 Kuroda, Y., Taira, Z., Uno, T. and Osaki, K., Cryst. Struct. Comm., 4, 321, (1975).

58 Kuroda, Y., Taira, Z., Uno, T. and Osaki, K., Cryst. Struct. Comm., 4, 325, (1975).

59 Stout, G.H., and Jensen L.H., "X-ray Structure Determination, a Practical Guide," The MacMillan Company, New York, 1968.

60 Jaffe, H.H. and Orchin, M., "Symmetry in Chemistry," John Wiley and Sons inc, New York, 1965.

61 Green, C.W., Call -apple, May 1983, 13, A.P.P.L.E. co-op, Renton, Washington.

62 De Groat, R., Call -apple, Feb 1981, 30, A.P.P.L.E. co-op, Renton, Washington.

63 Anastasi, A., "Psychological Testing," (fourth edition), The MacMillan Company, New York, 1976.

64 Heywood, J., "Assessment in Higher Education," John Wiley and Sons inc, New York, 1977.

ACKNOWLEDGEMENTS

There are many people I would like to thank for their help during the course of this project. In particular my thanks to my supervisors, Prof. B.R. Penfold and Dr. W.T. Robinson who were of great assistance throughout the course of my work. Thanks are also due to Dr. J.W. Blunt and R.D. Draper for their advice and the utility software they provided. I would like to acknowledge the assistance of the many people who aided me during the crystal structure determination, especially Dr. D.A. House who provided the crystals. I am indebted to L.N. Shustak for the kind loan of his printer for the printing of this thesis. Finally thanks to all the colleagues, friends, flatmates and family who have supported and encouraged me over the duration of my work.

ADDITIONAL MATERIAL

Included inside the back cover should be the following items:

1. Reprints of the following papers:

   Penfold, B.R. and Temple, R.S., J. Chem. Educ., 59, 775, (1982).

   Temple, R.S., House, D.A. and Robinson, W.T., Acta Cryst. C40, 1789, (1984).

2. Floppy disks (2) containing the following:

   KODE: The Symmetry Operations program disk.

   STAF: Symmetry Operations Questions disk 1.

   STAF2: Symmetry Operations Questions disk 2. (Reverse of KODE:).

   DOC: Symmetry Operations documentation disk. (Reverse of STAF:).

gave data which were at first interpreted as monoclinic.† It was only after full refinement that this structure turned out to be hexagonal (Wang, Gabe, Calvert & Taylor, 1977). It seems from the DDL report that their 'bloc sphérique' was not ground and furthermore no absorption correction was applied. In this case, the expected variations in absorption would be even larger and an orthorhombic distribution of intensities is possible.

The structure description and the chemical conclusions in DDL remain valid because they recognize the chemical similarity of the now symmetry-related Pb(1) and Pb(4) as well as Pb(2) and Pb(3). In fact, no deviations from the hexagonal structure were discussed by DDL: even the distortions shown in their Fig. 3 are hexagonal in nature and are confirmed by the present refinement.

Dr R. E. Marsh (private communication) raises the point that, in view of the non-singularity of the refinement in the *Cmcm* subgroup, it is disturbing that some Pb atoms in the orthorhombic refinement differ from their refined hexagonal positions by a number of σ's which is statistically highly significant, especially for the Pb(2) and Pb(3) atoms. To this valid statistical argument, we offer the following comments:

(*a*) All the atomic deviations between the MH refinement and the hexagonal model, except two, correspond to less than half the r.m.s. thermal-motion amplitude. The exceptions, atoms O(15) and O(16), deviate by about 0·1 Å, *i.e.* 0·8 r.m.s. amplitude or 4 positional σ's. If the orthorhombic distortion in MH is assumed to be correct, the room-temperature thermal motion would suffice to establish locally the hexagonal symmetry of the structure and consequently to produce alternate orientations of the orthorhombic structure in the metrically hexagonal lattice, leading to atomic disorder or twinning. The resulting diffraction intensities of a macroscopic sample would therefore mimic hexagonal symmetry.

† In this case, A^* varied by 30% (at $\theta = 15°$) for $r_1 = 0{\cdot}0063$ and $r_2 = 0{\cdot}0073$ cm respectively ($\mu\bar{r} = 2{\cdot}78$).

(*b*) In the MH refinement, Pb(3) behaves abnormally, and they suggest some disorder. In the hexagonal refinement, Pb(2) and Pb(3) occupy the same site which refines normally.

The normal distribution indicates that 11 out of 85 measurements of systematically absent intensities are expected to be larger than $1{\cdot}5\sigma$. This is consistent with the seven reported new systematic absences.

The metrically hexagonal lattice, the new class of systematic absences found in the DDL data, the satisfactory refinement in $P6_3/mcm$ and the disappearance of the anomaly at the Pb(3) position raise doubts about the space-group symmetries $Cmc2_1$ and *Cmcm* proposed for $Pb_3Mn_7O_{15}$ by DDL and MH respectively. We feel that this compound should be re-examined and the possibility that it is truly hexagonal or twinned should be considered by future investigators. If its orthorhombic character were to be confirmed, its structure should be described in terms of a very slight distortion of the above hexagonal structure. Unfortunately, no single crystal from the original preparation could be found by DDL and we were unable to attempt to synthesize it due to experimental difficulties.

We thank Drs B. Darriet, M. Devalette and B. Latourrette for their comments and for trying to find single crystals. We also thank Professor J. D. H. Donnay and Dr R. E. Marsh for their comments on a first version of this manuscript.

References

DARRIET, B., DEVALETTE, M. & LATOURRETTE, B. (1978). *Acta Cryst.* B**34**, 3528–3532.
FRIEDEL, G. (1916). *C. R. Acd. Sci.* **162**, 130–133.
MARSH, R. E. & HERBSTEIN, F. H. (1983). *Acta Cryst.* B**39**, 280–287.
WANG, Y., GABE, E. J., CALVERT, L. D. & TAYLOR, J. B. (1977). *Acta Cryst.* B**33**, 131–133.
WEBER, K. (1969). *Acta Cryst.* B**25**, 1174–1178.

Acta Cryst. (1984). C**40**, 1789–1791

trans-Aquachloro[(1*S*,4*S*,7*S*,8*R*,11*R*,14*R*)-5,5,7*e*,12,12,14*e*-hexamethyl-1,4,8,11-tetraazacyclotetradecane]chromium(III) Nitrate, $[CrCl(H_2O)(C_{16}H_{36}N_4)](NO_3)_2$

BY R. TEMPLE, D. A. HOUSE AND W. T. ROBINSON

Department of Chemistry, University of Canterbury, Christchurch, New Zealand

(*Received* 2 *April* 1984; *accepted* 13 *June* 1984)

Abstract. $M_r = 513{\cdot}96$, monoclinic, Cc, $a = 14{\cdot}560\ (5)$, $b = 11{\cdot}740\ (5)$, $c = 14{\cdot}772\ (7)$ Å, $\beta = 110{\cdot}83\ (3)°$, $Z = 4$, $V = 2360{\cdot}0$ Å^3, $D_x = 1{\cdot}447$ Mg m^{-3}, λ(Mo $K\alpha$) $= 0{\cdot}71069$ Å, $\mu = 0{\cdot}632$ mm^{-1}, $F(000) = 1092$, $T = 293$ K, final $R = 0{\cdot}0614$ for 1894 observed reflections. The macrocyclic ligand, teta, forms a CrN_4

0108-2701/84/111789-03$01.50

plane with Cl^- and H_2O occupying the *trans* positions. The *sec*-NH protons adopt the *meso* (*RSSR*) configuration and the methyl groups attached to C atoms at the 7 and 14 positions of the ligand adopt an equatorial orientation.

Introduction. The title compound is one of several possible isomers (House & Yang, 1982; House, Hay & Akbar Ali, 1983). It is formed as orange crystals when green *trans*-$[CrCl_2(teta)]ClO_4$ is allowed to hydrolyze in water, followed by the addition of $NaNO_3$. The crystal structure of this isomer allows the assignment of the stereochemistry of the coordinated *sec*-NH centers and the orientation of the methyl groups at the 7 and 14 positions of the macrocycle.

Experimental. Nicolet *R*3*m* automated four-circle diffractometer, graphite-monochromated Mo $K\alpha$. Crystal 0·25 × 0·21 × 0·16 mm. Lattice parameters from 25 reflections in range $28 < 2\theta < 32°$; space group *Cc* or *C*2/*c* from systematic absences; *Cc* confirmed from successful refinement. $\theta/2\theta$ scans, $2\theta_{max} = 50°$. Standard reflections (no variation) 002, $\bar{1}31$, $\bar{4}00$. 2367 reflections measured, 1894 with $I > 3\sigma(I)$ used in refinement, $h \pm 16$, k 0 to 13, l 0 to 17. No corrections for absorption or extinction. Coordinates of Cr obtained from Patterson calculations. 30 non-hydrogen atoms from difference Fourier maps. Blocked cascade least-squares refinement. All non-hydrogen atoms anisotropic, H atoms (except those of H_2O) included in calculated positions with isotropic thermal parameters equal to 1·2 times the isotropic equivalent of their carrier atoms (C—H = N—H = 0·96 Å); *F* magnitudes used; $R = 0{\cdot}0614$, $wR = 0{\cdot}0473$, $w = [|\sigma^2(F)|]^{-1}$, $S = 4{\cdot}597$, $(\Delta/\sigma)_{max} = 0{\cdot}595$ [C(5) *y* coordinate], $\Delta\rho = -0{\cdot}44$ to $+0{\cdot}64$ e Å^{-3}. All calculations on Nova 4X computer using *SHELXTL* (Sheldrick, 1981).

Discussion. Atomic parameters are listed in Table 1,* bond lengths and selected angles in Table 2. The crystal structure consists of independent six-coordinate $CrCl(teta)(H_2O)^{2+}$ cations (Fig. 1) and NO_3^- anions. The coordinated Cl^- and H_2O ligands are *trans* with respect to the CrN_4 plane and, within the macrocycle, the five-membered rings adopt the *gauche* ($\delta\lambda$) and the six-membered rings the chair (pp)† conformation. The *sec*-NH protons are in the lowest-energy *RSSR* (*meso*) configuration and the (*R*,*S*)C-methyl groups are in the

Table 1. *Atomic coordinates* ($\times 10^4$) *and equivalent isotropic temperature factors* (Å$^2 \times 10^3$)

	x	*y*	*z*	U_{eq}*
Cr	7500	2598 (1)	0	35
N(1)	6817 (6)	2346 (7)	1000 (7)	36
N(2)	6798 (5)	4171 (6)	−256 (5)	39
N(3)	8154 (4)	2813 (6)	−1031 (5)	37
N(4)	8192 (5)	998 (6)	278 (5)	35
C(1)	6436 (7)	1153 (8)	1059 (7)	45
C(11)	5925 (8)	1098 (10)	1785 (8)	68
C(2)	7296 (7)	338 (8)	1347 (7)	47
C(3)	7747 (8)	−29 (9)	565 (8)	50
C(31)	6919 (7)	−628 (9)	−285 (7)	71
C(32)	8576 (9)	−914 (9)	1075 (9)	69
C(4)	8513 (8)	817 (9)	−589 (8)	63
C(5)	8887 (7)	1875 (10)	−843 (8)	63
C(6)	8494 (6)	3998 (9)	−1108 (6)	46
C(61)	9044 (7)	4017 (10)	−1856 (7)	62
C(7)	7624 (7)	4793 (8)	−1391 (6)	46
C(8)	7181 (7)	5197 (8)	−636 (7)	43
C(81)	6325 (8)	5983 (9)	−1142 (9)	60
C(82)	7951 (6)	5839 (8)	212 (6)	48
C(9)	6507 (7)	4385 (8)	611 (6)	46
C(10)	6091 (6)	3287 (9)	842 (7)	45
Cl	8848 (2)	3284 (2)	1247 (2)	60
O(1)	6262 (5)	1968 (6)	−1105 (5)	51
N(5)	9002 (5)	2171 (7)	3520 (6)	55
O(2)	8139 (4)	2273 (7)	3232 (5)	70
O(3)	9530 (5)	3122 (6)	3924 (5)	59
O(4)	9465 (5)	1324 (7)	3485 (6)	81
N(6)	1068 (8)	2216 (9)	1597 (7)	75
O(5)	1534 (5)	2884 (7)	2277 (5)	76
O(6)	1127 (8)	2274 (9)	849 (7)	120
O(7)	583 (7)	1408 (7)	1789 (8)	113

* Equivalent isotropic *U* defined as one third of the trace of the orthogonalized U_{ij} tensor.

Table 2. *Bond distances* (Å) *and selected angles* (°)

Cr—Cl	2·307 (2)	Cr—N(1)	2·074 (11)	Cr—N(2)	2·079 (7)
Cr—N(3)	2·078 (8)	Cr—N(4)	2·102 (7)	Cr—O(1)	2·090 (6)
N(1)—C(1)	1·520 (13)	C(1)—C(11)	1·508 (17)	C(1)—C(2)	1·513 (3)
C(2)—C(3)	1·578 (17)	C(3)—C(31)	1·565 (13)	C(3)—C(32)	1·569 (15)
C(3)—N(4)	1·500 (14)	N(4)—C(4)	1·526 (15)	C(4)—C(5)	1·458 (17)
C(5)—N(3)	1·489 (13)	N(3)—C(6)	1·496 (13)	C(6)—C(61)	1·578 (16)
C(6)—C(7)	1·508 (13)	C(7)—C(8)	1·547 (17)	C(8)—C(81)	1·517 (14)
C(8)—C(82)	1·547 (12)	C(8)—N(2)	1·517 (13)	N(2)—C(9)	1·506 (13)
C(9)—C(10)	1·514 (14)	C(10)—N(1)	1·488 (13)	N(5)—O(2)	1·181 (9)
N(5)—O(3)	1·367 (10)	N(5)—O(4)	1·213 (11)	N(6)—O(5)	1·265 (12)
N(6)—O(6)	1·139 (15)	N(6)—O(7)	1·274 (15)		

N(3)—Cr—N(2)	94·7 (3)	N(3)—Cr—N(4)	86·5 (3)
N(2)—Cr—N(4)	178·8 (3)	N(3)—Cr—O(1)	87·9 (3)
N(2)—Cr—O(1)	86·6 (3)	N(4)—Cr—O(1)	93·3 (2)
N(3)—Cr—N(1)	178·2 (3)	N(2)—Cr—N(1)	85·8 (3)
N(4)—Cr—N(1)	93·0 (3)	O(1)—Cr—N(1)	90·5 (3)
N(3)—Cr—Cl	93·5 (2)	N(2)—Cr—Cl	93·3 (2)
N(4)—Cr—Cl	86·8 (2)	O(1)—Cr—Cl	178·6 (2)
N(1)—Cr—Cl	88·2 (2)		

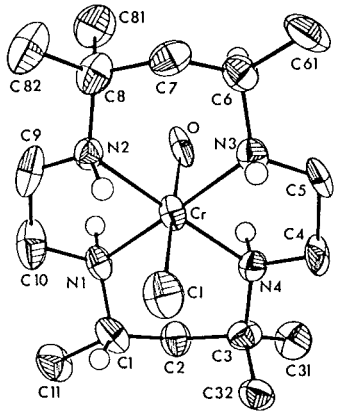


Fig. 1. A general view of the complex cation, *trans*-$CrCl(teta)(H_2O)^{2+}$.

* Lists of structure amplitudes, anisotropic thermal parameters, bond angles and H-atom coordinates have been deposited with the British Library Lending Division as Supplementary Publication No. SUP 39583 (15 pp.). Copies may be obtained through The Executive Secretary, International Union of Crystallography, 5 Abbey Square, Chester CH1 2HU, England.

† See Jurnak & Raymond (1972) for the nomenclature used here.

lowest-energy equatorial orientation. The conformation of the macrocycle is entirely similar to that found for the *trans*-$Cr(OCONH_2)_2$(cyclam) cation (cyclam is 1,4,8,11-tetraazacyclotetradecane) (Bang & Mønsted, 1982).

All bond lengths and angles within the cation are normal for such compounds (Bang & Mønsted, 1982) with Cr—N(mean), Cr—Cl and Cr—O distances of 2·083, 2·307 and 2·090 Å, respectively.

References

BANG, E. & MØNSTED, O. (1982). *Acta Chem. Scand. Ser. A*, **36**, 353–359.

HOUSE, D. A., HAY, R. W. & AKBAR ALI, M. (1983). *Inorg. Chim. Acta*, **72**, 239–245.

HOUSE, D. A. & YANG, D. (1982). *Inorg. Chim. Acta Lett.* **64**, L167–L169.

JURNAK, F. A. & RAYMOND, K. N. (1972). *Inorg. Chem.* **11**, 3149.

SHELDRICK, G. M. (1981). *SHELXTL User Manual*, 3rd revision. Nicolet XRD Corporation, Cupertino, California.

Acta Cryst. (1984). C**40**, 1791–1793

Structure of Bis(η^5-trimethylsilylcyclopentadienyl)titanium(IV) Pentasulfide, $[Ti(C_8H_{13}Si)_2]S_5$

BY NIKOLAOS KLOURAS AND STAVROS VOLIOTIS

Department of Chemistry, University of Patras, Patras, Greece

AND GABRIEL GERMAIN

Laboratoire de Chimie Quantique, Université de Louvain, 1348 Louvain-la-Neuve, Belgium

(*Received* 13 *June* 1983; *accepted* 16 *May* 1984)

Abstract. $M_r = 482·78$, monoclinic, $P2_1/c$, $a = 7·884$ (5), $b = 14·018$ (14), $c = 22·352$ (24) Å, $\beta = 96·28$ (7)°, $Z = 4$, $V = 2455·4$ (39) Å^3, $D_x = 1·30$ g cm^{-3}, $F(000) = 1008$, λ(Mo $K\alpha$) = 0·71069 Å, $\mu = 8·44$ cm^{-1}, room temperature, $R = 0·10$ for 2382 reflexions. The six-membered heterocycle TiS_5 has a chair conformation. The Ti atom has an irregular tetrahedral environment comprising the bidentate pentasulfide fragment [Ti—S = 2·416 (5) and 2·450 (5) Å] and the two centroids (R's) of the substituted cyclopentadienyl rings (Ti—R = 2·076 and 2·059 Å).

Introduction. The comparison of the structure of $Ti(\eta^5\text{-}C_5H_5)_2Cl_2$ with that of $Ti(\eta^5\text{-}C_5H_4CH_3)_2Cl_2$ demonstrates that the substitution of the H atom by a CH_3 group on each cyclopentadienyl ring has a negligible effect on the basic molecular configuration. Specifically, the corresponding bond distances and angles differ by less than 0·01 Å and 1·3°, respectively (Petersen & Dahl, 1975). Moreover, the two cyclopentadienyl rings are symmetrically disposed with respect to the TiS_2 fragment in $Ti(\eta^5\text{-}C_5H_5)_2(SC_6H_5)_2$: The TiS_2 plane approximately bisects the R—Ti—R angle (Epstein, Bernal & Köpf, 1971; Muller, Petersen & Dahl, 1976). In contrast, no symmetrical positioning of the $Ti(\eta^5\text{-}C_5H_5)_2$ fragment relative to TiS_2 (which is part of the TiS_5 ring) has been observed in $Ti(\eta^5\text{-}C_5H_5)_2S_5$ (II). The normal to the TiS_5 plane forms angles of 32 and 17° with the Ti—R ring vectors.

The purpose of the present work is to elucidate the disposition of the two substituted cyclopentadienyl rings relative to the six-membered TiS_5 ring system in the title compound (I).

Experimental. The title compound was prepared by the reaction of $Ti[\eta^5\text{-}C_5H_4Si(CH_3)_3]Cl_2$ (Köpf & Klouras, 1982) with excess $(NH_4)_2S_5$ in methanol, using a method similar to that described by Köpf & Block (1969) and Köpf & Kahl (1974). After recrystallization from $CHCl_3/CH_3OH$ (1:1), dark-red glistening crystals were obtained. The compound is air-stable, m.p. 382 K. Composition: found: C = 39·72, H = 5·31 wt% [$M_r = 482$ (mass spectrum)]; calculated: C = 38·80, H = 5·43 wt%. ^{1}H NMR peaks were found at τ(p.p.m.) 3·97 (*m*) and 4·17 (*m*) (C_5H_4, relative intensity 4), 9·77 (*s*) and 9·84 (*s*) [$Si(CH_3)_3$, relative intensity 9].

Syntex diffractometer, 2θ range 0–47°, graphite-monochromatized Mo $K\alpha$ radiation. 3807 reflections measured; after averaging, 2382 independent observed reflections [$I > 2·5\sigma(I)$] remained. Intensities not corrected for absorption or extinction. Structure solved by direct methods using *MULTAN*80 (Main, Fiske, Hull, Lessinger, Germain, Declercq & Woolfson, 1980), and refined by blocked full-matrix least squares using *SHELX* (Sheldrick, 1976); all atoms except H considered to be anisotropic. All H atoms placed in geometrically calculated positions. Refinement converged with unit weights to $R = 0·1034$.

0108-2701/84/111791-03$01.50

to zero. The resulting secular determinant looks like Figure 2a of the paper by Sims and Ewing. It can be converted to a 2 × 2 block diagonal form by regrouping the basis functions into odd and even sets. In this form, four energy values are obtained by solving two quadratic equations. The corresponding normalized wave functions are obtained by the usual procedure. The results are:

$$W_1 = 4.979\ (4.935)\text{: } \phi_1 = 1.007f_1 - 0.0219f_3$$
$$W_2 = 19.76\ (19.74)\text{: } \phi_2 = 1.044f_2 + 0.135f_4$$
$$W_3 = 55.02\ (44.41)\text{: } \phi_3 = 0.336f_1 - 1.062f_3$$
$$W_4 = 100.2\ (78.96)\text{: } \phi_4 = 0.287f_2 + 1.074f_4$$

The students will note that increasing the trial function from one to four terms has improved the ground-state energy to 4.979 hartrees (0.9% error). The essential features of the variational method can be stressed further by comparing the correct eigenfunctions ψ_n, with the basis function, f_n, and the variational function, ϕ_n, for each value of n.

Better approximations to the correct energy eigenvalues and eigenfunctions can be obtained by increasing the number of terms in the trial wave function. An odd (f_5) and even (f_6) function can be added and six energy eigenvalues can be obtained by solving two cubic equations. The secular determinant and the resulting solutions are given below.

$$\begin{vmatrix}
5.00 - 1.00E & 2.64 - 0.336E & 2.31 - 0.112E & 0 & 0 & 0 \\
2.64 - 0.336E & 50.0 - 1.00E & 51.3 - 0.563E & 0 & 0 & 0 \\
2.31 - 0.112E & 51.3 - 0.563E & 157.3 - 1.00E & 0 & 0 & 0 \\
0 & 0 & 0 & 21.0 - 1.00E & 17.2 - 0.383E & 16.1 - 0.243E \\
0 & 0 & 0 & 17.2 - 0.383E & 94.6 - 1.00E & 108.9 - 0.707E \\
0 & 0 & 0 & 16.1 - 0.243E & 108.9 - 0.707E & 239.7 - 1.00E
\end{vmatrix} = 0$$

$$W_1 = 4.974\text{: } \phi_1 = 1.005f_1 - 0.014f_3 - 0.007f_5$$
$$W_2 = 19.75\text{: } \phi_2 = 1.046f_2 - 0.147f_4 + 0.010f_6$$
$$W_3 = 48.59\text{: } \phi_3 = 0.356f_1 - 1.196f_3 + 0.274f_5$$
$$W_4 = 80.04\text{: } \phi_4 = 0.280f_2 - 1.339f_4 + 0.444f_6$$
$$W_5 = 175.3\text{: } \phi_5 = 0.032f_1 - 0.463f_3 + 1.185f_5$$
$$W_6 = 285.1\text{: } \phi_6 = 0.050f_2 - 0.629f_4 - 1.344f_6$$

At this point, things start to get a bit tedious and the students begin to look forlornly at the computer. They are now ready to appreciate the versatile computer program described by Sims and Ewing.

The details of the calculation described in this paper are available on request from the author.

Introducing Microcomputers into an Analytical Lab Course

Brian J. Pankuch
Union College
Cranford, NJ 07016

Recently, we developed a series of programs which we used as a basis for introducing microcomputers into an analytical lab course. Our aim was for programs that would:

1. Accept student data and still give output (purchased programs frequently worked with instructor data but not with student data).
2. Provide a forgiving, friendly atmosphere and the opportunity for a student to correct input mistakes, run the program, check the results on a CRT, and obtain printed copy of the output.
3. Have the raw data in the compiled program on a diskette so the author of the program can thoroughly debug the program or make "improvements" and test the "improvements" immediately with real data.

Our programs—written for the titration of monobasic and dibasic acids—will accept a student's experimental data, graph the titration curves on the screen, and print out the results of all calculations. Each student has a diskette that contains the UCSD PASCAL operating system with the data reduction program transferred to it. With the easy-to-use UCSD PASCAL operating system it is quite straightforward for the student to proceed and exchange the normality of base actually used for the one already in the program; similar manipulations are done with sample weight and data pairs of ml and pH.

This operating system is very forgiving of any errors. One can simply move the cursor to the error and type in the correction. Once the data is typed in, the student is asked to have the data checked by someone else, and the program is then compiled.

Once the program is running with the student's data, the output is examined in tabular form, and both a rough and smoothed graphical form. The instructor generally examines this output and, if it is acceptable, the student is sent over to a second microcomputer which is configured differently to provide printed output.

Students who have had no previous experience with computers are able to titrate at least two trials of unknown acid, completely debug their program with their data in it, and obtain hard copy of the results during one six-hour period. Most are able to run only one data set through the computer. Students are required to graph all results, calculate the pK's and percent composition, and identify the acid present by traditional methods.

The above requires one Apple connected to a CRT for every seven or eight students, and one additional Apple interfaced with a printer for the entire class.

The results from the experiment have been quite satisfactory. Both traditional and computer calculations have agreed, making it possible to identify the unknown from the pK's and to calculate the composition in all cases. Each time the lab is run new ideas are suggested, and many are worth implementing. Having a number of copies of the program with real laboratory data incorported makes testing any changes made in the program much more realistic and faster. We have found the UCSD PASCAL language and operating system very helpful in meeting our aims.

Program TITRATION—UCSD Apple PASCAL Version 1.1. The program is 500 lines, self-documenting, used with a 64K Apple with language card. Documentation includes diskette with program for the titration of a dibasic acid (student titration data in program), step-by-step directions for actual use in lab. For the diskette and documentation, send a check for $10 made out to Dr. Brian J. Pankuch.

Space Group Generation and Display Using Pascal

B. R. Penfold and R. S. Temple
University of Canterbury
Christchurch 1, New Zealand

Some knowledge of space group theory and a general familiarity with space group diagrams is essential for an understanding of structure in the crystalline state, whether or not the student is specifically concerned with structure analysis. Most advanced students find it hard to visualize in three dimensions the results of individual space group operations and even harder to appreciate the product of several operations. While the building of three-dimensional models, as described by Hathaway (*3*) is undoubtedly a powerful teaching aid, it is quite time-consuming and only a very limited number of space groups can be dealt with by any one student.

In order to provide experience of a wide range of space

groups, with many repetitions if necessary, but without excessive use of time, we have written a program, SGROUP, in the UCSD PASCAL language, to generate and display space group diagrams on a microcomputer graphics screen. The program is appropriate for final-year degree students and may be used either by an instructor, for class teaching, or by an individual student for extended practice. The diagrams are developed one operation at a time and are displayed in the standard format of "International Tables for X-ray Crystallography" (Vol. I) (*4*). Any orthorhombic space group and any monoclinic space group may be generated as a single display which includes both symmetry operation symbols and equivalent positions.

On executing the program, the user is first presented with a list of the available commands, e.g., SCR to generate a twofold screw operation, INV to generate an inversion through a point. The list may be displayed at any time by invoking the HELP command. Full prompting is provided to aid complete space group generation. The example shown in Figure 1 is of space group $Pna2_1$ which was generated by the following sequence of user responses (in italics) to a series of computer prompts, assuming the default values for x, y, z (see below) have been selected. (The unit cell remains displayed throughout the entire dialogue.)

REF, GLI, SCR, ROT, INV. CEN, HELP, STAT, END
SCR
WHICH AXIS IS THE SCREW AXIS (A-B-C)
C
INPUT THE COORDINATES OF THE AXIS
0,0 (new position generated by screw axis displayed)
REF, GLI, SCR, ROT, INV, CEN, HELP, STAT, END
GLI
INPUT AXIS NORMAL TO GLIDE (A-B-C)
A
WHICH AXIS IS THE TRANSLATION ALONG
—B—C—N—
N
INPUT FRACTIONAL COORDINATE OF PLANE
0.25 (new positions generated by *n* glide displayed)
REF, GLI, SCR, ROT, INV, CEN, HELP, STAT, END
GLI
INPUT AXIS NORMAL TO GLIDE (A-B-C)
B
WHICH AXIS IS THE TRANSLATION ALONG
—A—C—N—
A
INPUT FRACTIONAL COORDINATE OF PLANE
0.25 (*a* glide plane symbol displayed but no new positions generated)
REF, GLI, SCR, ROT, INV, CEN, HELP, STAT, END
END

The content of the final display matches closely the corresponding diagram in "International Tables," but the fact that it is developed in a stepwise manner gives it a valuable additional teaching function. The effect of each symmetry operation is graphically demonstrated, each equivalent position being clearly identified with a particular operation by being flashed for several seconds when it is generated.

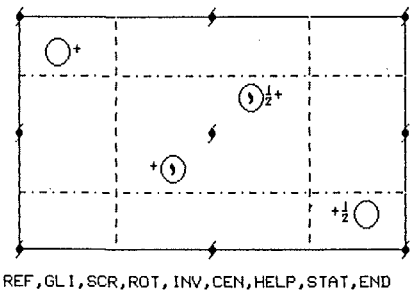


Figure 1. Screen display of space group $Pna2_1$ showing prompt line. (Penfold)

The computer's inherent flexibility provides for numerous teaching possibilities. For example, one particularly valuable feature is the ability to display three alternative views of each space group simply by interchanging axes. Many views of space groups which are not included in "International Tables" may thus be generated. A very useful student exercise is in fact to transform the conventional view to an alternative view. The most common mistakes will be associated with glide planes, and these will show up as inconsistencies between the sets of equivalent positions displayed in the two views. Another example is the demonstration of the generation of a center of inversion by combination of a plane (mirror or glide) with an axis (rotation or screw) normal to it.

The coordinates of the reference point in the unit cell (general position x, y, z) are set, by default, to match the "International Tables" diagrams. They may, however, be optionally varied by the user. This feature can be used, for example, to illustrate how the separation of certain equivalent positions varies with the proximity of the reference point to a nontranslational operation. The coordinates of the reference point could indeed be chosen to be those of a special position which, upon application of all the group operations, would be shown to be a position of reduced multiplicity. Figure 2 illustrates such an option where, in space group $Pmc2_1$ the value of the x coordinate was chosen as 0.5.

An extension beyond the "International Tables" format is a provision for choosing a group of atoms rather than just a single circle as the repeating unit of structure. This is of particular value in conveying the idea of an asymmetric unit in the crystal structure and the relation between molecular symmetry and crystal symmetry. It is sometimes convenient in this "molecule" mode to use the program option to suppress the display of symmetry symbols. The example illustrated in Figure 3 in space group $Pmc2_1$ was generated by the following sequence of responses:

DO YOU WANT MOLECULES INSTEAD OF CIRCLES
YES
HOW MANY ATOMS DO YOU WANT
MAX = 7
4

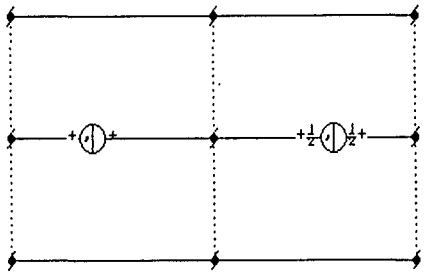


Figure 2. Screen display of space group $Pmc2_1$ illustrating special position of point symmetry *m*. (Penfold)

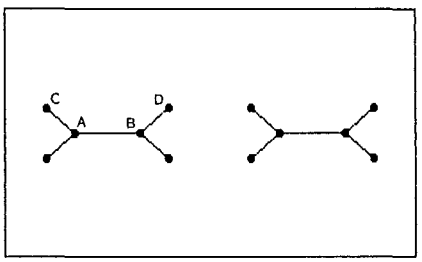


Figure 3. Screen display on "molecule" mode for space group $Pmc2_1$ with display of symmetry symbols suppressed. Each of the atoms in the group A, B, C, D (the asymmetric unit) is entered with specific coordinates. (Penfold)

INPUT THE X, Y, Z COORDINATES OF ATOM 1
.5,.17,.1
HOW MANY BONDS DO YOU WANT EXTENDING FROM ATOM 1
2
WHICH ATOM DOES BOND 1 GO TO
2
WHICH ATOM DOES BOND 2 GO TO
3
INPUT THE X, Y, Z COORDINATES OF ATOM 2
.5,.31,.1
HOW MANY BONDS DO YOU WANT EXTENDING FROM ATOM 2
1 (i.e., the existing bond from ATOM 1 not to be included)
WHICH ATOM DOES BOND 1 GO TO
4
INPUT THE X, Y, Z, COORDINATES OF ATOM 3
0.4,.1,.1
HOW MANY BONDS DO YOU WANT EXTENDING FROM ATOM 3
0 (i.e., all such bonds specified already)
INPUT THE X, Y, Z COORDINATES OF ATOM 4
.4,.38,.1
HOW MANY BONDS DO YOU WANT EXTENDING FROM ATOM 4
0 (asymmetric unit displayed on screen)
ARE THE OPERATIONS TO BE DISPLAYED (Y/N)
N
REF, GLI, SCR, ROT, INV, CEN, HELP, STAT, END
SCR
WHICH AXIS IS THE SCREW AXIS
C
INPUT THE COORDINATES OF THE AXIS
0,0 (the second asymmetric unit displayed)
REF, GLI, SCR, ROT, INV, CEN, HELP, STAT, END
GLI
INPUT AXIS NORMAL TO GLIDE (A-B-C)
B
WHICH AXIS IS THE TRANSLATION ALONG
—A—C—N—
C
INPUT FRACTIONAL COORDINATE OF PLANE
0 (the remaining asymmetric units displayed)
REF, GLI, SCR, ROT, INV, CEN, HELP, STAT, END
END

The program has been used successfully both as a student demonstration in a class setting and by individual students working at their own pace, preferably following a class demonstration. In the latter situation it may be desirable for the student to have access to the "International Tables" or a selection from them.

Program SGROUP was written in Apple-UCSD PASCAL. Execution requires 48 Kbytes and the Apple language system (including one or more disk drives) on an Apple II or Apple II+ microcomputer. The resolution of the graphics display is 280 × 192 points. Execution of the program on any other UCSD PASCAL system should require only modification of the constants relating to the graphics display. Copies of the program, both source and object, are available from the authors on 5.25-in. diskette and/or as a listing. Send money order for $5 made out to Department of Chemistry, University of Canterbury, Christchurch 1, New Zealand, and, if necessary, a blank diskette.

Estimation of A_∞ in Reaction-Rate Studies

John J. Houser
The University of Akron
Akron, OH 55325

A perennial problem in the measurement of rate constants is the determination of the "infinity reading," A_∞, where a long reaction half-life or the presence of subsequent reactions may render the A_∞ value difficult to measure or of questionable validity. Figure 4 shows the consequence of using an A_∞ that is either too large or too small. Calculation of infinity values can eliminate these problems.

Wiberg (*5*) has described a program for calculating first-order rate constants given t,A_t data and an A_∞, in which the A_∞ is optimized by minimizing the root-mean-square deviation between experimental and recalculated data points. In the method of Moore (*6*), a nonlinear least-squares treatment for first-order reactions, the sum of the squares of the deviations is minimized, and the "best" A_0, A_∞, and k are calculated. We have developed a simpler method for computing A_∞ which does not require an initial experimental value. While our program, called CONINFIN, is currently set up for first-order and for second-order reactions in one component, it in principle is applicable to reactions of any order, though a separate algorithm would have to be devised for each type. Our procedure, a variation of the direct grid search method (*7*), uses the correlation coefficient, R, as a measure of the goodness of fit of the $t,f(A_\infty - A_t)$ data to the appropriate rate law. Though an equivalent criterion would have been the standard deviation of the slope, σ_s, and Wiberg has shown (*5b*) that the percent error in the slope, $\%\epsilon_s$, is often a more sensitive measure than R of nonlinearity in the plot, we felt that R, being constrained in all cases to vary from 0 to ±1, is a more immediately recognizable criterion of linearity. Based on our tests, described below, only rate data of exceptional precision would benefit from the increased sensitivity of the $\%\epsilon_s$ minimization method. CONINFIN accepts as input the presumed maximum and minimum values of A_∞, the number of steps between these values, the number of iterations to be made, the order of the reaction and the t,A_t data points. The program proceeds stepwise from the maximum to the minimum value of A_∞, at each step calculating a set of $\ln(A_\infty - A_t)$ or $1/(A_\infty - A_t)$ data, depending on the order. The correlation coefficient, R_i, is then calculated for the linear least-squares regression of $f(A_\infty - A_t)_{ij}$ on t_j. The next value of A_∞ is calculated and the process repeated. After the first pass from maximum to minimum A_∞, the value which yielded the largest R is incremented and decremented by 10% and a new set of R_i is calculated within this range. Further passes narrow the range of A_∞ to any desired degree. In most cases, four to six passes will yield a value having more significant figures than are usable, and further refinement is therefore unnecessary. Once the "best" A_∞ value has been found, a final set of $t,f(A_\infty - A_t)$ data is calculated and stored in a file or punched on cards. A standard linear regression program is then used to

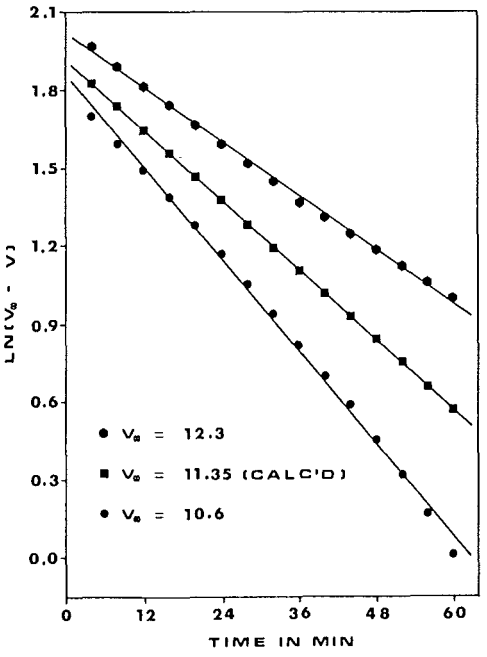


Figure 4. First-order rate plots using estimated and calculated infinity values. (Houser)